```
                                                                                      B 1
VNTI - VNP NETWORK INITIALIZATI MACRO V05.03b Saturday 29-Jun-85 02:46  Page 110-4
Symbol table

MPCSR = 177746        OF$OFF= 000002        OP$SIN= 000010        PS.PIC= 000100        SBPOOL= ****** GX
MRKFL   000040R   002 OF$ON = 000000        OP$STA= 000004        PS.SYS= 000040        SBUF    001776R
MULP    002536R   003 OF$SMC= 000200        OP$TPA= 000020        PUMR  = ****** GX     SCHEL   001140R   003
MUXP    002530R   003 OF.PRO= 000040        OP$TRI= 000040        P$$P45= 000000        SCHELL  000442R   003
MXCP    000064R   002 OF.RLU= 000100        OP$UCS= 000020        P$$WRD= 000000        SCHEL0  001130R   003
M$$CRB= 000124        OF.SMC= 000200        OP$USE= 000002        P.BLKS  000016        SCHEL1  000434R   003
M$$CRX= 000000        ONAME   000032R   002 OP$VEC= 000040        P.BUSY  000024        SCTIM   001116R   003
M$$FCS= 000000        OPNCFG  005560R       OP$VER= 000040        P.IOC   000003        SCTIMM  000424R   003
M$$MGE= 000001        OP$ADD= 000200        OP$XAC= 000001        P.LNK   000000        SEG     011220R
M$$NET= 000000        OP$ALL= 000200        OP$XPR= 000002        P.MAIN  000012        SERADD  000140RG  002
M$$OVR= 000000        OP$CCS= 000004        OP$X5S= 000004        P.NAM   000004        SERALL  014640R
M.BFVH  000011        OP$CIE= 004000        OP$X9S= 000010        P.OWN   000026        SERBEG  000110RG  002
M.BFVL  000012        OP$CON= 000400        OUTT    010[illegible]52R       P.PRI   000002        SERCIR  000116RG  002
M.LGTH= 000014        OP$COP= 000001        O.FLG   000003        P.REL   000014        SERCR   015356RG
M.LNK   000000        OP$COS= 000100        O.LEN   000012        P.SIZE  000016        SERCUR  000256RG  002
M.UMRA  000002        OP$DST= 000200        O.LNK   000000        P.STAT  000030        SERDCT  000136RG  002
M.UMRN  000004        OP$DTE= 000040        O.MXC   000004        P.SUB   000010        SERDEV  000120RG  002
M.UMVH  000010        OP$DUP= 000002        O.NAM   000006        P.SWSZ  000022        SERDF   002170R   003
M.UMVL  000006        OP$EST= 000100        O.TYP   000002        P.TCB   000026        SERDMP  000132RG  002
NA      000076R   002 OP$EVE= 010000        PARSE   006232R       P.WAIT  000020        SERDV   014414RG
NAMBUF  000022R   002 OP$FIL= 000200        PAS     015204RG      QF$ACT= 000001        SEREND  014620RG
NAME    000030R   002 OP$GRP= 000100        PBLOX = ****** GX     QF$ALL= 000002        SERFIN  000266RG  002
NAMED   000000R   002 OP$HOS= 000001        PDVNAM= ****** GX     QF$KNO= 000004        SERFLG  000110RG  002
NBEAJ   000106R   002 OP$HTM= 000002        PIRQ  = 177772        QF$LOO= 000010        SERHST  000114RG  002
NBRA    000102R   002 OP$INA= 004000        PMODE = 030000        QINS    017272R       SERINI  014532RG
NC.FM0= 000000        OP$INC= 001000        PREFIX  001734R       QIO     002006R       SERMUL  014314RG
NC.FM1= 000001        OP$KDS= 002000        PRELEN= 000016        QUICK   014102R       SERMUX  014324RG
NC.FM2= 000002        OP$KDT= 000400        PREWLN= 000024        Q$$OPT= 000010        SERNOD  000112RG  002
NN      000074R   002 OP$KEV= 001000        PREWRN  001752R       Q.IOAE= 000012        SEROFF  000260RG  002
NOAD    014354RG      OP$KGR= 001000        PRODF   000712R   003 Q.IOEF= 000006        SERPAS  000122RG  002
NOAR    014334RG      OP$KNT= 010000        PRODF1  000750R   003 Q.IOFN= 000002        SERSLN  014460RG
NODC    004620R       OP$KSK= 002000        PROFLG  000720R   003 Q.IOLU= 000004        SERSLT  000262RG  002
NODCT   010420R       OP$LCT= 000200        PRONAM  001730R   003 Q.IOPL= 000014        SERST   002520R   003
NODDF   001266R   003 OP$LNE= 000040        PROTXT  006256R       Q.IOPR= 000007        SERTMP  000146RG  002
NONMAP= ****** GX     OP$LOC= 000001        PRVUNM  000002R   002 Q.IOSB= 000010        SERUNI  014304RG
NS$DON= 000000        OP$LST= 000002        PR0   = 000000        RAD     000104R   002 SETBLK  016040R
NS$SDI= 000002        OP$LTM= 000001        PR1   = 000040        RBRA  = 000076        SETST   016674RG
NS$WDC= 000004        OP$LVL= 000010        PR2   = 000100        REMARE  000100R   002 SF.ACT= 000200
NTIERR  001410R       OP$MAC= 000010        PR3   = 000140        REMDF   001522R   003 SF.CIR= 000010
NTPNAM= ****** GX     OP$MCO= 000002        PR4   = 000200        REQNTL  020140R       SF.DEV= 000020
NTPPAR= ****** GX     OP$MDE= 000400        PR5   = 000240        RETF    011146R       SF.DIA= 004000
NTPSAV= ****** GX     OP$MLI= 000004        PR6   = 000300        RF.LOO= 100000        SF.DPA= 000100
NTPSCN  005606R       OP$MON= 000100        PR7   = 000340        ROUDF   001632R   003 SF.DPC= 000200
NTPSIZ= ****** GX     OP$MST= 000020        PS    = 177776        RQB   = ****** GX     SF.DUM= 010000
NTWARN  001554R       OP$NAM= 000004        PSTART= ****** GX     R$$DER= 000000        SF.ENA= 000100
N$$ACC= 000001        OP$NET= 004000        PS.APR= 000007        R$$K11= 000001        SF.HAD= 000004
N$$BUF= 000001        OP$NLI= 000002        PS.CHK= 010000        R$$SND= 000000        SF.HST= 000002
N$$LDV= 000001        OP$NOD= 000020        PS.CKP= 040000        R$$11M= 000000        SF.LOA= 002000
N$$MCP= 000001        OP$OUT= 002000        PS.CKR= 020000        R.ADD   000010        SF.LPB= 000004
N$$MLL= 000001        OP$OWN= 000020        PS.COM= 000200        R.FLAG  000012        SF.MFL= 000040
N$$MOV= 000010        OP$PAR= 000010        PS.DEL= 000010        R.LEN   000014        SF.PAC= 000020
N$$NCT= 000001        OP$PRI= 000010        PS.DRV= 000020        R.LNK   000000        SF.PH3= 020000
N$$PEM= 000001        OP$RET= 010000        PS.FXD= 004000        R.NAM   000002        SF.PSS= 000040
OBJDF   00134[illegible]R   003 OP$RPA= 000010        PS.LIO= 001000        SBFLG = ****** GX     SF.REA= 000010
OBJ[illegible]   001366R   003 OP$SCD= 000001        PS.NSF= 000400        SBNAM = ****** GX     SF.SEC= 000400
OF$INS= 000001        OP$SEG= 000400        PS.OUT= 100000        SBPAR = ****** GX     SF.SER= 000001
OF$LOG= 000100        OP$SER= 000400        PS.PER= 002000        SBPCB = ****** GX     SF.SVC= 000002
```

```
                                                                                      C 1
VNTI - VNP NETWORK INITIALIZATI MACRO V05.03b Saturday 29-Jun-85 02:46  Page 110-5
```

```
                                                                     B  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50     PAGE 11
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE               REFERENCES

OP$INC  = 001000            #5-78
OP$KDS  = 002000            #5-78
OP$KDT  = 000400            #5-78
OP$KEV  = 001000            #5-78
OP$KGR  = 001000            #5-78
OP$KNT  = 010000            #5-78
OP$KSK  = 002000            #5-78
OP$LCT  = 000200            #5-78
OP$LNE  = 000040            #5-78
OP$LOC  = 000001            #5-78         #5-78
OP$LST  = 000002            #5-78
OP$LTM  = 000001            #5-78
OP$LVL  = 000010            #5-78
OP$MAC  = 000010            #5-78
OP$MCO  = 000002            #5-78
OP$MDE  = 000400            #5-78
OP$MLI  = 000004            #5-78
OP$MON  = 000100            #5-78
OP$MST  = 000020            #5-78
OP$NAM  = 000004            #5-78         #5-78         #5-78
OP$NET  = 004000            #5-78
OP$NLI  = 000002            #5-78
OP$NOD  = 000020            #5-78
OP$OUT  = 002000            #5-78
OP$OWN  = 000020            #5-78
OP$PAR  = 000010            #5-78
OP$PRI  = 000010            #5-78
OP$RET  = 010000            #5-78
OP$RPA  = 000010            #5-78
OP$SCO  = 000001            #5-78
OP$SEG  = 000400            #5-78
OP$SER  = 000400            #5-78
OP$SIN  = 000010            #5-78
OP$STA  = 000004            #5-78
OP$1PA  = 000020            #5-78
OP$TRI  = 000040            #5-78
OP$UCS  = 000020            #5-78
OP$USE  = 000002            #5-78
OP$VEC  = 000040            #5-78
OP$VER  = 000040            #5-78
OP$XAC  = 000001            #5-78
OP$XPR  = 000002            #5-78
OP$X5S  = 000004            #5-78
OP$X9S  = 000010            #5-78
OUTT      010552 R          #67-3044
O.FLG     000003            *86-3682
O.LEN     000012             86-3674       86-3686       87-3726
O.MXC     000004            *86-3683
O.NAM     000006            *86-3680      *86-3681
O.TYP     000002            *86-3679       86-3691       87-3727
PARSE     006232 R           30-1769       30-1772       30-1775       30-1784       30-1799       30-1812      #30-1842
PAS       015204 RG         #95-4095
```

```
                                                                     C  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50     PAGE 12
```

```
MACRO CROSS REFERENCE                                  CREF    04.00

MACRO NAME        REFERENCES

                  #76-3396    76-3396     #76-3408    76-3408     #76-3420    76-3420     #79-3506    79-3506     #80-3542    80-3542
                  #86-3686    86-3686     #87-3726    87-3726     #88-3747    88-3747     #88-3762    88-3762     #94-4060    94-4060
                  #94-4063    94-4063     #94-4076    94-4076     #98-4258    98-4258     #98-4272    98-4272     #102-4451   102-4451
                  #102-4454   102-4454    #102-4467   102-4467    #103-4524   103-4524    #103-4526   103-4526    #103-4530   103-4530
                  #103-4536   103-4536    #105-4707   105-4707    #105-4709   105-4709
RESRG             #5-58       97-4236      98-4266    98-4285      102-4437   102-4469     103-4537   105-4714
RETURN             8-227      8-528        9-564      10-607       11-745     12-802       13-845     14-890       15-924     16-1022
                   17-1104    24-1329      24-1332    24-1335      24-1338    24-1344      24-1348    24-1376      24-1453    24-1459
                   24-1485    24-1489      25-1500    25-1503      25-1506    25-1509      26-1627    26-1639      28-1709    28-1713
                   29-1731    30-1837      30-1848    31-1876      50-2542    51-2557      52-2573    54-2612      55-2626    56-2646
                   56-2648    57-2661      58-2671    59-2678      59-2715    60-2726      60-2737    60-2740      61-2780    62-2799
                   62-2802    62-2823      62-2826    63-2846      63-2849    63-2870      63-2873    64-2898      65-2957    65-2963
                   65-2966    66-3006      67-3021    67-3031      67-3040    67-3049      67-3059    67-3068      67-3077    67-3086
                   67-3096    67-3106      67-3113    68-3141      68-3146    69-3194      69-3200    69-3216      69-3219    70-3237
                   70-3245    71-3264      71-3268    72-3289      72-3293    73-3307      73-3310    74-3332      74-3335    75-3359
                   75-3362    75-3378      75-3381    76-3398      76-3401    76-3422      76-3425    77-3451      78-3471    79-3515
                   80-3547    81-3566      82-3589    83-3615      84-3633    85-3653      86-3696    86-3701      87-3730    88-3767
                   89-3791    89-3799      90-3829    90-3860      90-3863    91-3880      91-3883    91-3886      91-3892    91-3895
                   91-3901    91-3904      91-3910    91-3915      91-3918    92-3951      92-3971    93-4006      93-4013    94-4079
                   95-4131    96-4145      96-4166    96-4195      97-4238    98-4267      98-4287    99-4324      99-4347    100-4371
                   101-4414   102-4439     102-4471   103-4555     103-4576   103-4602     104-4633   104-4679     105-4715   106-4762
                   107-4793   108-4837     109-4865   110-4884
RNBDF$            #5-57       5-64
SAVRG             #5-58       97-4211      98-4256    102-4431     103-4521   105-4701
SCBDF$            #5-56       5-76
SERDF$            #5-58       5-81
SLTDF$            #5-55       5-73
SOB                59-2697
STATE$            #5-56        18-1110    #18-1112    #18-1114    #18-1116    #18-1118    #18-1120    #18-1122    #18-1124    #18-1126
                  #18-1128    #18-1130    #18-1132    #18-1134     19-1141    #19-1143    #19-1145    #19-1147    #19-1149    #19-1151
                  #19-1153    #19-1155    #19-1157    #19-1159    #19-1161    #19-1163    #19-1165    #19-1167    #19-1169    #19-1171
                  #19-1173     20-1180    #20-1182    #20-1185    #20-1187    #20-1190    #20-1192    #20-1194    #20-1197    #20-1200
                  #20-1203    #20-1206    #20-1209    #20-1211    #20-1214    #20-1216     21-1222    #21-1225    #21-1228    #21-1231
                  #21-1236    #21-1239    #21-1242    #21-1245     22-1252    #22-1254    #22-1256    #22-1258    #22-1261     23-1267
                  #23-1269    #23-1271    #23-1273    #23-1275    #23-1277    #23-1279    #23-1281    #23-1284    #23-1289    #23-1292
                  #23-1294    #23-1296    #23-1298    #23-1301    #23-1304    #23-1307    #23-1310    #23-1313    #23-1316    #23-1319
                  #23-1321     32-1890    #32-1909     33-1916    #33-1919    #33-1922     34-1929    #34-1931    #34-1933    #34-1939
                  #34-1941    #34-1944    #34-1946    #34-1949     35-1956    #35-1958    #35-1960    #35-1962    #35-1964    #35-1971
                  #35-1973    #35-1975    #35-1977    #35-1980    #35-1983    #35-1986    #35-1989    #35-1992    #35-1996    #35-1999
                   36-2009    #36-2011    #36-2013    #36-2015    #36-2018     37-2030     38-2038    #38-2040    #38-2042    #38-2044
                  #38-2046    #38-2048    #38-2050    #38-2052    #38-2054    #38-2056    #38-2058    #38-2060    #38-2062     39-2070
                  #39-2073    #39-2076    #39-2080    #39-2083    #39-2086    #39-2089    #39-2092    #39-2095    #39-2098    #39-2102
                  #40-2112    #40-2115    #40-2118    #40-2121    #40-2124    #40-2127    #40-2130    #40-2133    #40-2136    #40-2139
                  #40-2142    #40-2145    #40-2148    #40-2151    #40-2154    #40-2157    #40-2160    #40-2163    #40-2166    #40-2169
                  #40-2172    #40-2175     41-2184    #41-2187    #41-2190    #41-2193    #41-2196    #41-2199    #41-2202     42-2209
                  #42-2212    #42-2215    #42-2218    #42-2221    #42-2224    #42-2232    #42-2235     43-2243     44-2251    #44-2253
                  #44-2255    #44-2257    #44-2259    #44-2261    #44-2263    #44-2265    #44-2267    #44-2269    #44-2271    #44-2273
                  #44-2275    #44-2277    #44-2279    #44-2281    #44-2283    #44-2285     45-2294    #45-2302    #45-2304    #45-2306
                  #45-2312    #45-2315    #45-2317     46-2332    #46-2337    #46-2341    #46-2345    #46-2349    #46-2353    #46-2357
                  #46-2361    #46-2365    #46-2368    #47-2373    #47-2378    #47-2381    #47-2384    #47-2390    #47-2393     48-2401
                  #48-2404    #48-2409    #48-2412    #48-2416    #48-2420    #48-2424    #48-2427    #48-2431    #48-2435    #48-2439
                  #48-2442    #48-2445    #48-2449    #48-2453    #48-2457    #49-2466    #49-2470    #49-2474    #49-2478    #49-2482
```

```
    369
    370 000440  010100                      50$:    MOV     R1,R0           ; COPY PARTITION ADDRESS
    371 000442  001402                              BEQ     60$             ; IF NONE - BRANCH
    372 000444                                      CALL    $XPAR           ; RELEASE PARTITION
    373 000450  016700  000152'             60$:    MOV     TMPPCB,R0       ; COPY MICROCODE PARTITION ADDRESS
    374 000454                                      CALL    $XPAR           ; DEALLOCATE PARTITION
    375
    376 000460  005001                              CLR     R1              ; INDICATE NO PARTITION
    377 000462  016700  000250'                     MOV     ERROR,R0        ; STORE ERROR CODE
    378 000466  000261                      70$:    SEC                     ; INDICATE ERROR
    379 000470                                      RETURN
    380
    381 000472  032767  000020  000000G 80$:    BIT     #FE.PLA,$FMASK  ; PLAS SUPPORT ?
    382 000500  001416                              BEQ     85$             ; IF NO - BRANCH
    383 000502                                      GETRV   R1,.PLGTH       ; READ PCB
    384 000520  016702  000000G                     MOV     .PPRO,R2        ; STORE PARTITION PROTECTION OFFSET
    385 000524  012762  177777  000000G             MOV     #-1,$BFR(R2)    ; NO PLAS ACCESS
    386 000532                                      PUTRC                   ; WRITE PCB TO IMAGE
    387 000536  016767  000152' 000252' 85$:    MOV     TMPPCB,PCBA     ; STORE MICROCODE PCB ADDRESS
    388 000544  016701  000252'                     MOV     PCBA,R1         ; ...
    389
    390 000550  000241                      90$:    CLC                     ; SUCCESS
    391 000552                                      RETURN
    392
    393 000554  005067  000250'             100$:   CLR     ERROR           ; DO NOT TYPE GENERAL CASE ERROR
    394 000560                                      ERRPT$  UNAOP,MSGPRI    ; TYPE LOCAL ERROR
    395 000570  000413                              BR      120$
    396 000572                              110$:   CALL    $CLDEQ          ; CLOSE FILE
    397 000576  016777  000000G 000000G             MOV     $IOSB,@$CLIOS   ; COPY FILE ERROR CODE
    398 000604  005067  000250'                     CLR     ERROR           ; DO NOT TYPE GENERAL CASE ERROR
    399 000610                                      ERRPT$  UNARE,MSGPRI    ; TYPE READ ERROR
    400 000620  005001                      120$:   CLR     R1              ; INDICATE NO MICROCODE PARTITION SET
    401 000622  000706                              BR      50$             ; CONTINUE
    402
    917                                            .SBTTL  FNDPDV - FIND A PDV ADDRESS
    918                                    ;+
    919                                    ; FNDPDV - TRY TO FIND PDV
    920                                    ;
    921                                    ; INPUTS:
    922                                    ;       R0=RAD50   V NAME
    923                                    ;
    924                                    ; OUTPUTS:
    925                                    ;       C-BIT=SUCCESS/FAILURE
    926                                    ;       R0=PDV ADDRESS
    927                                    ;       R1=PDV INDEX
    928                                    ;       R2=DESTROYED
    929                                    ;-
    930 003100                             FNDPDV:
    931 003100  010046                             MOV     R0,-(SP)                ; SAVE NAME TO BE MATCHED
    932 003102  016701  000000G                    MOV     $PDVTA,R1               ; GET PDV TABLE ADDRESS
    933 003106  016702  000000G                    MOV     $PDVNM,R2               ; AND NUMBER OF PDV'S
    934 003112  001407                             BEQ     30$                     ; IF ZERO, NO PDV'S!
    935 003114  012100                     10$:    MOV     (R1)+,R0                ; GET PDV ADDRESS
    936 003116  001403                             BEQ     20$                     ; IF ZERO, SKIP THIS ONE
    937 003120  026016  000000G            15$:    CMP     Z.NAM(R0) (SP)          ; DOES THE NAME MATCH?
    938 003124  001403                             BEQ     40$                     ; IF EQ, YES (C-BIT CLEAR)
    939 003126  005302                     20$:    DEC     R2
    940 003130  001371                             BNE     10$                     ; LOOP $PDVNM TIMES
    941 003132  000261                     30$:    SEC                             ; INDICATE FAILURE
    942 003134  005226                     40$:    INC     (SP)+                   ; PURGE STACK
    943 003136  006016                             ROR     (SP)                    ; SAVE C-BIT
    944 003140  005741                             TST     -(R1)                   ; CALCULATE PDV INDEX
    945 003142  166701  000000G                    SUB     $PDVTA,R1               ;
    946 003146  006316                             ASL     (SP)                    ; RESTORE C-BIT
    947 003150                                     RETURN
    948
    949
    950                                    ;+
    951                                    ; MSGPRI - TYPE ERROR MESSAGE
    952                                    ;
    953                                    ; INPUTS:
    954                                    ;       R2 - ARGUMENT BLOCK ADDRESS
    955                                    ;
    956                                    ; OUTPUTS:
    957                                    ;       TYPE MESSAGE
    958                                    ;-
    959
    960 003152                             MSGPRI: CALL    $SAVAL                  ; SAVE REG'S
    961 003156  012700  000262'                    MOV     #MBUF,R0                ; GET ERROR BUFFER ADDRESS
    962 003162                                     CALL    $ERMSG                  ; FORMAT ERROR
    963 003166                                     CALL    $TIOUT                  ; TYPE MESSAGE
    964
    965 003172                                     RETURN
    966
    967         000001                             .END
```

PRMP
PROM

PRMP

```
     75                                          .SBTTL  PROMPTS
     76                                  ;
     77                                  ; INITIALIZE TABLE PSECTS
     78                                  ;
     79 000000                                   ISTAT$  X,Y
     80                                  ;
     81                                  ; DEFINE PROMPTS
     82                                  ;
     83 000000                                   PRMPT$  ^\''VNP>''\,P$VNP,<:>
     84
     85 000000                                   PRMPT$  <''Account: ''>,                P$ACC,<:>
     86 000000                                   PRMPT$  <''Active polling ratio: ''>,   P$ACT,<:>
     87 000000                                   PRMPT$  <''Address: ''>,                P$ADD,<:>
     88 000000                                   PRMPT$  <''Alias name: ''>,             P$AID,<:>
     89 000000                                   PRMPT$  <''Circuit identification: ''>, P$CRID,<:>
     90 000000                                   PRMPT$  <''Clear circuit options: ''>,  P$CLCI,<:>
     91 000000                                   PRMPT$  <''Clear component: ''>,        P$CLE,<:>
     92 000000                                   PRMPT$  <''Clear line options: ''>,     P$CLLI,<:>
     93 000000                                   PRMPT$  <''Clear logging options: ''>,  P$CLLO,<:>
     94 000000                                   PRMPT$  <''Clear node options: ''>,     P$CLNO,<:>
     95 000000                                   PRMPT$  <''Clear object options: ''>,   P$CLOB,<:>
     96 000000                                   PRMPT$  <''Cost: ''>,                   P$COS,<:>
     97 000000                                   PRMPT$  <''Component type: ''>,         P$PEN,<:>
     98 000000                                   PRMPT$  <''Copies: ''>,                 P$COP,<:>
     99 000000                                   PRMPT$  <''CSR address: ''>,            P$CSR,<:>
    100 000000                                   PRMPT$  <''Dead polling ratio: ''>,     P$DEA,<:>
    101 000000                                   PRMPT$  <''Destination node: ''>,       P$DES,<:>
    102 000000                                   PRMPT$  <''Destination: ''>,            P$DST,<:>                      ;[TD001]
    103 000000                                   PRMPT$  <''DTE address: ''>,            P$DTE,<:>
    104 000000                                   PRMPT$  <''Duplex: ''>,                 P$DUP,<:>
    105 000000                                   PRMPT$  <''Event identification: ''>,   P$EVE,<:>
    106 000000                                   PRMPT$  <''Group name: ''>,             P$GRP,<:>
    107 000000                                   PRMPT$  <''Host node identification: ''>, P$HOS,<:>
    108 000000                                   PRMPT$  <''Hello timer (sec): ''>       P$HTM,<:>
    109 000000                                   PRMPT$  <''Line identification: ''>,    P$LNID,<:>
    110 000000                                   PRMPT$  <''Listen timer (sec): ''>      P$LTM,<:>
    111 000000                                   PRMPT$  <''Line state: ''>,             P$LIST,<:>
    112 000000                                   PRMPT$  <''Location in memory: ''>,     P$LOC,<:>
    113 000000                                   PRMPT$  <''Logging sink type: ''>,      P$LGID,<:>
    114 000000                                   PRMPT$  <''Logging state: ''>,          P$LOST,<:>
    115 000000                                   PRMPT$  <''Maximum option: ''>,         P$MAX,<:>
    116 000000                                   PRMPT$  <''Maximum controllers: ''>,    P$MXCO,<:>
    117 000000                                   PRMPT$  <''Maximum lines: ''>,          P$MXLI,<:>
    118 000000                                   PRMPT$  <''Module identification: ''>,  P$MOID,<:>
    119 000000                                   PRMPT$  <''Multipoint ratio type: ''>,  P$MUL,<:>
    120 000000                                   PRMPT$  <''Node address: ''>,           P$NAD,<:>
    121 000000                                   PRMPT$  <''Network: ''>,                P$NET,<:>                      ;[TD001]
    122 000000                                   PRMPT$  <''Node identification: ''>,    P$NID,<:>
    123 000000                                   PRMPT$  <''Node name: ''>,              P$NAM,<:>
    124 000000                                   PRMPT$  <''Node state: ''>,             P$NOST,<:>
    125 000000                                   PRMPT$  <''Object name: ''>,            P$OBNA,<:>
    126 000000                                   PRMPT$  <''Object number: ''>,          P$OID,<:>
    127 000000                                   PRMPT$  <''Object user: ''>,            P$OBUS,<:>
    128 000000                                   PRMPT$  <''Object verification: ''>,    P$OBVE,<:>
    129 000000                                   PRMPT$  <''Owner name: ''>,             P$OWN,<:>
    130 000000                                   PRMPT$  <''Partition name: ''>,         P$PAR,<:>
    131 000000                                   PRMPT$  <''Priority: ''>,               P$PRI,<:>
    291 000214  160102                          SUB     R1,R2           ; CALCULATE LENGTH OF FREE SPACE
    292 000216  000405                          BR      40$             ; CONTINUE IN COMMON CODE
    293                                 ;
    294                                 ; UNMAPPED SYSTEMS
    295                                 ;
    296 000220  160102                  30$:    SUB     R1,R2           ; CALCULATE LENGTH OF FREE SPACE
    297 000222  005701                          TST     R1              ; WAS THERE A LISTHEAD ?
    298 000224  001402                          BEQ     40$             ; IF EQ, NO
    299 000226  066701  000000C                 ADD     $BFR+P.RFL,R1   ; POINT AT START OF FREE SPACE
    300                                 ;
    301                                 ; COMMON CODE FOR MAPPED AND UNMAPPED
    302                                 ;
    303 000232  010160  000000G         40$:    MOV     R1,Z.AVL(R0)    ; INITIALIZE LISTHEAD
    304 000236  001430                          BEQ     50$             ; IF EQ, NO LISTHEAD - LEAVE
    305                                 ;
    306                                 ; UPDATE FIRST BLOCK IN PROCESS PARTITION
    307                                 ;
    308 000240                                  GETAD   R1,#4,$BIAS     ; READ IN TWO WORDS POINTED TO BY THE LISTHEAD
    309 000264  005067  000000G                 CLR     $BFR            ; ZERO THE LINK POINTER
    310 000270  010267  000002G                 MOV     R2,$BFR+2       ; STORE THE FREE SPACE BYTE COUNT
    311 000274                                  PUTAD   R1,#4,$BIAS     ; WRITE OUT THE BLOCK POINTER
    312
    313 000320                          50$:    CALL    AXSYM           ; FILL IN SPECIAL SYMBOLS IF PROCESS IS AUX
    314 000324  103402                          BCS     70$             ; IF CS, AUX TIMER CELLS MISSING
    315 000326                                  RETURN
    316
    317                                 ;
    318                                 ; ERROR CONDITIONS
    319                                 ;
    320 000330                          60$:    CRASH                   ; $XFER SHOULD NOT HAVE BEEN CALLED
    321 000332                          70$:    EMSG$R  QX              ; AUX TIMER CELLS MISSING
    322 000340  016000  000230'         TSKERR: MOV     ETAB2-2(R0),R0  ; GET ERROR MESSAGE CONTROL BLOCK ADDRESS
    323 000344                                  CALLR   @(R0)+          ; PRINT ERROR MESSAGE
    324
    911 003110  012704  000000G         20$:    MOV     #$BFR,R4                ; POINT TO NODE COUNTERS
    912 003114  010005                          MOV     R0,R5                   ; SAVE COUNTERS SIZE
    913 003116  006300                          ASL     R0                      ; CONVERT TO BYTES
    914 003120  005024                  25$:    CLR     (R4)+                   ; INIT XPT NODE COUNTERS
    915 003122                                  SOB     R5,25$                  ; IF MORE - BRANCH
    916
    917 003126                                  PUTAD   P1,R0,R2                ; WRITE XPT NODE COUNTER BLOCK
    918 003146                          30$:
    919
    920 003146  016700  000000G                 MOV     $PDVA,R0                ; GET XPT PDV ADDRESS
    921 003152  005005                          CLR     R5                      ; INIT REG
    922 003154  156005  000000G                 BISB    Z.LLN(R0),R5            ; GET XPT CHANNELS
    923 003160  016002  000000G                 MOV     Z.DAT(R0),R2            ; GET XPT DDB ADDRESS
    924
    925 003164  005305                          DEC     R5                      ; DON'T INCLUDE CHANNEL 0
    926 003166  010500                          MOV     R5,R0                   ; COPY NUMBER OF CIRCUITS FOR $MUL
    927 003170  001511                          BEQ     50$                     ; IF EQ, NO LINES IN SYSTEM
    928 003172                                  GETRV   R2,#N$XLEN              ; READ IN XPT'S DDB
    929 003210  010567  000100G                 MOV     R5,N$TLC+$BFR           ; SAVE IN DATA DESCRIPTOR
    930 003214                                  PUTRC                           ; WRITE XPT DDB
    931
    932 003220  012701  000046                  MOV     #T$LLEN,R1              ; GET COUNTER BLOCK SIZE
    933 003224                                  CALL    $MUL                    ; CALCULATE TOTAL SIZE
    934 003230  062701  000077                  ADD     #77,R1                  ; ROUND UP TO NEXT BLOCK
    935 003234  042701  000077                  BIC     #77,R1                  ; ...
    936 003240                                  ASR$    6,R1                    ; DETERMINE NUMBER OF BLOCKS
    937
    938 003254  010204                          MOV     R2,R4                   ; SAVE XPT'S DDB ADDRESS
    939 003256                                  CALL    $ALN18                  ; ALLOCATE POOL..
    940 003262  103455                          BCS     101$                    ; IF ALLOCATION FAILURE - BRANCH
    941 003264  010402                          MOV     R4,R2                   ; RESTORE XPT'S DDB ADDRESS
    942 003266                                  GETRV   R2,#N$XLEN              ; READ XPT'S DDB
    943
    944 003304  010004                          MOV     R0,R4                   ; SAVE COUNTER BIAS
    945 003306  010067  000102G                 MOV     R0,$BFR+N$TLC+2         ; STORE COUNTER BIAS
    946 003312  005000                          CLR     R0                      ; SET UP VIRTUAL ADDRESS
    947 003314                                  BIAS    R0                      ; SET FOR APR6
    948 003324  010067  000104G                 MOV     R0,$BFR+N$TLC+4         ; STORE VIRTUAL ADDRESS
    949 003330                                  PUTRC                           ; WRITE OUT XPT'S DDB
    950
    951 003334                          40$:    GETAD   R0,#T$LLEN,R4           ;READ IN COUNTER BLOCK
    952 003356  012702  000000G                 MOV     #$BFR,R2                ;POINT TO COUNTER BLOCK AREA
    953 003362  012722  177777                  MOV     #-1,(R2)+               ;MARK LINE COUNTER BLOCK AS FREE
    954 003366  012703  000022                  MOV     #<<T$LLEN/2>-1>,R3      ;GET REMAINING SIZE (IN WORDS)
    955
    956 003372  005022                  45$:    CLR     (R2)+                   ;ZERO REMAINDER OF BLOCK
    957 003374                                  SOB     R3,45$                  ;...
    958 003400                                  PUTAD                           ;REWRITE UPDATED COUNTER BLOCK
    959 003404  062700  000046                  ADD     #T$LLEN,R0              ;POINT TO NEXXT COUNTER BLOCK
    960
    961 003410                                  SOB     R5,40$                  ;INITIALISE ALL BLOCKS
    962
    963 003414                          50$:
    964 003414                                  RETURN
    965
    966 003416                          101$:   EMSG$R  9D                      ; CIRCUIT COUNTER ALLOCATION FAILURE
    967
```

```
                                                                    B  9
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05      PAGE 3
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE             REFERENCES

LS.PRO  = 000002          #5-76
LS.PWF  = 040000          #5-76
LS.TOP  = 002000          #5-76
LS.UNF  = 020000          #5-76
LS.X25  = 000012          #5-76
LS.11D  = 004000          #5-76
LX.CEX  = 000004          #5-76
LX.LIN  = 000006          #5-76
LX.PRO  = 000005          #5-76
L.COST    000015          #5-72
L.CTL     000012          #5-72
L.CVA     177776          #5-72
L.DDM     000002          #5-72
L.DDS     000004          #5-72
L.DLC     000003          #5-72
L.DLM     000006          #5-72
L.DLS     000010          #5-72
L.FLG     000000          #5-72      11-531
L.KRBA    000016          #5-72
L.LEN   = 000022          #5-72
L.MPF     000022          #5-72
L.NMST    000020          #5-72
L.NSTA    000014          #5-72      11-534
L.OWNR    000021          #5-72
L.UNT     000013          #5-72
MILBLK    000140 R         6-177    #6-181      9-394
M$$MGE  = 000000           12-647    15-947     16-1018    17-1117    17-1127    17-1140
N$ADJ1    000072          *14-836
N$ADJ2    000074          *14-846
N$CACH    000062          *13-722    13-724    *13-789    *13-790
N$ELEN    000054           17-1111   17-1141
N$ENC     000042          *17-1112  *17-1118   *17-1142
N$MHC1    000036          *16-1013  *16-1016   *16-1019    16-1038    16-1039    16-1040
N$MHC2    000044          *16-1025  *16-1026   *16-1027   *16-1030   *16-1031   *16-1032    16-1066    16-1068    16-1069
N$ROA1    000022          *12-641   *12-645    *12-648     12-655     12-657
N$ROA2    000030          *12-654   *12-655    *12-658
N$TLC     000100           11-573    11-574     11-576    *15-929    *15-945    *15-948
N$TNC     000114           15-878    15-879
N$XLEN    000124           11-571    12-640     13-720     13-788     14-835     14-845     15-877     15-928     15-942
                           16-1010   16-1065
N$$VCT  = ******           8-350
PBLK      000032 R        #6-134     10-478     10-483
PRSYM     000600 R         7-253    #10-413
PR1SC     000162 R        #6-187      9-397
PR100M    000150 R        #6-183      9-396
PSYM      000036 R        #6-136    *10-464    *10-467
PVAL      000042 R        #6-137      7-276      7-288
P.BLKS  = ******  GX       7-272
P.REL   = ******  GX       7-284      7-299
R$$EIS  = ******           7-289      7-290     12-633     15-936     16-1003    17-1105
R$$MPL  = ******           5-62       6-98       6-130      7-262     10-427     19-1197
R$$11D  = ******           7-289      7-290     12-633     15-936     16-1003    17-1105
```

```
                                                                    C  9
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05      PAGE 4
```

```
    141
    142                                 .EVEN
    143
    144 000716  000000                  $GENST::.WORD   0               ; GEN PARTITION START ADDRESS
    145 000720  000000                  $GENND::.WORD   0               ; GEN PARTITION END ADDRESS
    622 003416                   $PFLAG::.BLKW   1           ; PSI characteristics word                      ;[TD001]
    623           000001         PF.XDF==        1           ; System has extended data formats              ;[TD001]
    624           000002         PF.PSV==        2           ; System contains PSV (gateway) module          ;[TD001]
    625           000004         PF.XGA==        4           ; System contains XGA (host access) module      ;[TD001]
    626                          ;
    627                          ; END OF TASK BUFFER POINTERS
    628                          ;
    629 003420                   $XBUF2::.BLKW   1           ; ADDRESS OF REMAINDER OF XXBUF
    630 003422                   $XLEN2::.BLKW   1           ; LENGTH OF BINARY BUFFER
    631                          ;
    632                          ; LOAD REQUEST BLOCK
    633                          ;
    634 003424                   $$LRB:: .BLKW   2           ; REQUESTING TASK'S NAME
    635 003430                   RQB::   .BLKW   40.         ; INPUT PARAMETERS
    636           003430'        RQBST== RQB+LR.STS
    637
    638                          ;
    639                          ; AREA SUPPORT PARAMETERS
    640                          ;
    641           176000         ARAMSK==176000
    642           000001         LEVL1==1
```

$GETAD - GET RECORD FROM SYSTEM IMAGE FILE

```
   1224 004764   103402                         BLO     65$              ; BR IF NO
   1225 004766   042716  160000                 BIC     #160000,(SP)     ; CLEAR MAPPING BITS
   1226 004772   062603                 65$:    ADD     (SP)+,R3         ; COMPUTE OFFSET ( VA + BLOCK REMAINDER )
   1227 004774   010301                         MOV     R3,R1            ; COPY OFFSET
   1228 004776   042703  177000                 BIC     #177000,R3       ; CONVERT TO OFFSET INTO 512. BYTE BLOCK
   1229 005002   105001                         CLRB    R1               ; ALMOST CLEAR OFFSET INTO BLOCK
   1230 005004   000301                         SWAB    R1               ; ALIGN BLOCKS OF OFFSET WITH STARTING
   1231 005006   006201                         ASR     R1               ;  BLOCK
   1232 005010   060102                         ADD     R1,R2            ; FORM STARTING BLOCK OF OFFSET
   1233 005012   062702  000003                 ADD     #3,R2            ; FORM VIRTUAL BLOCK NUMBER WITHIN FILE
   1234 005016   005001                         CLR     R1               ; CLEAR HIGH ORDER VBN FOR .POINT
   1235 005020   000657                         BR      27$
   1236
   1237                                         .DSABL  LSB
```

```
                                                           B 13
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16     PAGE 4
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE               REFERENCES

LR.ACO    000052             #5-80
LR.ADD    000012             #5-80
LR.ALI    000002             #5-80
LR.BLK    000010             #5-80
LR.CTL    000004             #5-80
LR.DES    000010             #5-80
LR.HLP    000002             #5-80
LR.LIN    000002             #5-80      5-80       5-80       5-80       5-80       5-80       5-80
LR.NAM    000012             #5-80
LR.PAS    000041             #5-80
LR.PRO    000002             #5-80
LR.STS    000000             #5-80      15-636
LR.TRI    000006             #5-80
LR.TYP    000002             #5-80
LR.UCB    000016             #5-80
LR.UIC    000020             #5-80
LR.UNT    000005             #5-80
LS.ALI  = 000016             #5-80
LS.CEX  = 000001             #5-80
LS.CIR  = 000033             #5-80
LS.CXO  = 010000             #5-80
LS.CX5  = 000400             #5-80
LS.ECH  = 001000             #5-80
LS.FDX  = 004000             #5-80
LS.HDX  = 010000             #5-80
LS.HLP  = 000012             #5-80
LS.LIN  = 000003             #5-80
LS.LMC  = 000007             #5-80
LS.LOG  = 000020             #5-80
LS.MOD  = 000036             #5-80
LS.NOD  = 000014             #5-80
LS.NTI  = 000013             #5-80
LS.OBJ  = 000022             #5-80
LS.OPT  = 000400             #5-80
LS.PRO  = 000002             #5-80
LS.TOP  = 002000             #5-80
LS.TST  = 000010             #5-80
LS.UNF  = 020000             #5-80
LS.XIT  = 000011             #5-80
LX.ALI  = 000017             #5-80
LX.CEX  = 000004             #5-80
LX.CIR  = 000034             #5-80
LX.LIN  = 000006             #5-80
LX.LOG  = 000021             #5-80
LX.NOD  = 000015             #5-80
LX.OBJ  = 000023             #5-80
LX.PRO  = 000005             #5-80
L.COST    000015   G         #5-78
L.CTL     000012   G         #5-78
L.CVA     177776   G         #5-78
L.DDM     000002   G         #5-78
L.DDS     000004   G         #5-78
```

```
                                                           C 13
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16     PAGE 5
```

```
**FILE**ID**VRQB






VV      VV   RRRRRRRR       QQQQQQ     BBBBBBBB
VV      VV   RRRRRRRR       QQQQQQ     BBBBBBBB
VV      VV   RR      RR   QQ      QQ   BB      BB
VV      VV   RR      RR   QQ      QQ   BB      BB
VV      VV   RR      RR   QQ      QQ   BB      BB
VV      VV   RR      RR   QQ      QQ   BB      BB
VV      VV   RRRRRRRR     QQ      QQ   BBBBBBBB
VV      VV   RRRRRRRR     QQ      QQ   BBBBBBBB
VV      VV   RR  RR       QQ  QQ  QQ   BB      BB
VV      VV   RR  RR       QQ  QQ  QQ   BB      BB
  VV  VV     RR    RR     QQ    QQ     BB      BB       ....
  VV  VV     RR    RR     QQ    QQ     BB      BB       ....
    VV       RR      RR     QQQQ  QQ   BBBBBBBB         ....
    VV       RR      RR     QQQQ  QQ   BBBBBBBB         ....






LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
VRQB    VND PROCESS REQUEST BLOC MACRO V05.03b Tuesday 30-Jul-85 10:53
```

```
                                              .SBTTL  CHKNOD - CHECK NODE PARMS
                                      ;+
                                      ; CHKNOD - CHECK NODE PARAMETERS
                                      ;
                                      ; INPUTS:
                                      ;       $OPTON = NODE OPTIONS SELECTED
                                      ;
                                      ; OUTPUTS:
                                      ;       C-BIT=SUCCESS/FAILURE
                                      ;
                                      ;-

000234                                CHKNOD:
000234  032767  000000G 000000G               BIT     #VF.PEM,$VFLAG          ; IS THIS PROTCL EMLTR CMD ?
000242  001073                                BNE     30$                     ; IF NE, YES .. SKIP CHECKS
000244  032767  000172  000000G               BIT     #OP$NLI!OP$RPA!OP$TPA!OP$VER!OP$EST,$OPTON ; XPT NEEDED ?
000252  001467                                BEQ     30$                     ; IF EQ, NO
000254  012700  114224                        MOV     #^RXPT,R0               ; SEARCH FOR XPT'S
000260                                        CALL    $FPDV                   ; ..PDV ADDRESS
000264  103464                                BCS     101$                    ; IF CS, XPT NOT FOUND
000266  005760  000000G                       TST     Z.PCB(R0)               ; IS PROCESS LOADED ?
000272  001464                                BEQ     111$                    ; IF EQ, NO

000274  032767  000002  000000G               BIT     #OP$NLI,$OPTON          ; IS THIS A LOOPBACK NODE REQUEST ?
000302  001453                                BEQ     30$                     ; IF EQ, NO .. OKAY
000304  032767  000000G 000000G               BIT     #CM$SET,$CMAND          ; IS THIS A SET REQUEST ?
000312  001447                                BEQ     30$                     ; IF EQ, NO
000314  032767  000000G 000000G               BIT     #VF.ADD,$VFLAG          ; WAS NODE ID GIVEN AS ADDRESS ?
000322  001061                                BNE     141$                    ; IF NE, YES .. ERROR
000324  016746  000246'                       MOV     PDVX,-(SP)              ; SAVE PDV INDEX FOR XPT
000330                                        CALL    SLTA                    ; ELSE, GET SLT ADDRESS
000334  012603                                MOV     (SP)+,R3                ; GET XPT PDV INDEX
000336  103436                                BCS     40$                     ; IF CS, LINE NOT FOUND
000340  016700  000000G                       MOV     $LLCTA,R0               ; POINT AT REVERSE TABLE
000344  016701  000000G                       MOV     $SLN,R1                 ; GET LINE NUMBER
000350  006301                                ASL     R1                      ; CONVERT TO WORDS
000352  060100                                ADD     R1,R0                   ; POINT AT ENTRY IN REVERSE TABLE
000354  016701  000000G                       MOV     $SLTA,R1                ; GET SLT ADDRESS
000360  005710                                TST     (R0)                    ; IS REVERSE TABLE POINTING AT STATIONS ?
000362  100404                                BMI     10$                     ; IF MI, YES
                                      ;
                                      ; PT-TO-PT LINE
                                      ;
000364  105761  000000G                       TSTB    L.NSTA(R1)              ; IS THE LINE MULTIPOINT ?
000370  001030                                BNE     121$                    ; IF NE, YES .. ERROR
000372  000411                                BR      20$                     ; ELSE, CONTINUE
                                      ;
                                      ; MULTIPOINT LINE
                                      ;
000374  105761  000000G               10$:    TSTB    L.NSTA(R1)              ; IS THIS LINE MULTIPOINT ?
000400  001424                                BEQ     121$                    ; IF EQ, NO .. ERROR
000402  011000                                MOV     (R0),R0                 ; GET ADDRESS/2 OF STATION TABLE
000404  006300                                ASL     R0                      ; CONVERT TO STATION TABLE ADDRESS
000406  016702  000000G                       MOV     $TRIB,R2                ; GET TRIBUTARY NUMBER
000412  006302                                ASL     R2                      ; CONVERT TRIBUTARY NUMBER TO WORD INDEX
000414  060200                                ADD     R2,R0                   ; POINT AT PDV INDEX AND CHANNEL NUMBER

```

```
                                                .SBTTL  CPRO - CHECK CLEAR PROCESS PARMS
                                        ;+
                                        ; CPRO - CHECK "CLEAR PROCESS" REQUEST BLOCK
                                        ;
                                        ; THIS ROUTINE IS CALLED TO CHECK THAT A CLEAR PROCESS REQUEST IS VALID. IF
                                        ; THE PROCESS IS AN LLC, ALL ITS CHANNELS MUST BE INACTIVE AND, IF IT IS ALSO
                                        ; AN RSX11M PSEUDO-DEVICE, THE DEVICE MUST BE DISMOUNTED. IF THE PROCESS IS
                                        ; EITHER A DLC OR A DDM, ALL LINES WHICH INCLUDE THE PROCESS MUST BE NOT READY.
                                        ;
                                        ; INPUTS:
                                        ;       $PDVA=PDV ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       C-BIT=SUCCESS/FAILURE
                                        ;       R4 CLEARED IF PROCESS IS NON-RESIDENT
                                        ;-
001672  005000                          CPRO:   CLR     R0                      ; USED AS ERROR FLAG
001674  016701  000000G                         MOV     $PDVA,R1                ; GET PDV ADDRESS
001700  005761  000000G                         TST     Z.PCB(R1)               ; IS PROCESS CORE RESIDENT?
001704  001003                                  BNE     10$                     ; IF NE, NO
001706  005004                                  CLR     R4                      ; INDICATE NO WORK TO DO
001710  000167  000554                          JMP     150$

001714  022767  004640  000000G 10$:    CMP     #^RAUX,$NAME            ; IS IT AUX ?
001722  001003                                  BNE     14$                     ; IF NE, NO
001724                                          EMSG$R  ZR                      ; PROCESS MAY NOT BE UNLOADED

001732  032767  000000G 000000G 14$:    BIT     #FL.LLC,$FLAGS          ; IS THIS AN LLC?
001740  001002                                  BNE     19$                     ; IF YES - BRANCH
001742                                          CALLR   100$                    ; CONTINUE

001746  022767  114224  000000G 19$:    CMP     #^RXPT,$NAME            ; IS IT XPT ?
001754  001020                                  BNE     15$                     ; IF NO BRANCH

001756  016702  000000G                         MOV     $SLTMA,R2               ; GET SLT TABLE ADDRESS
001762  016746  000000G                         MOV     $SLTNM,-(SP)            ; GET NUMBER OF LINES
001766  012203                          5$:     MOV     (R2)+,R3                ; GET NEXT SLT
001770  032763  000000G 000000G                 BIT     #LF.RDY,L.FLG(R3)       ; IS LINE LOADED ?
001776  001404                                  BEQ     4$                      ; IF NO - BRANCH
002000  005726                                  TST     (SP)+                   ; CLEAN UP STACK
002002                                          EMSG$R  ZB                      ; LINE IN WRONG STATE

002010  005316                          4$:     DEC     (SP)                    ; COUNT ONE LESS SLT
002012  001365                                  BNE     5$
002014  005726                                  TST     (SP)+                   ; CLEAN UP STACK

002016  000241                          15$:    CLC                             ; JUST IN CASE
002020  005002                                  CLR     R2                      ; GET CHANNEL COUNT
002022  156102  000000G                         BISB    Z.LLN(R1),R2            ;
002026  001446                                  BEQ     40$                     ; IF ZERO, NONE TO CHECK
002030  006302                                  ASL     R2                      ; CONVERT TO A WORD INDEX
002032  062701  000000G                         ADD     #Z.MAP,R1               ; POINT AT MAPPING TABLE
002036  060102                                  ADD     R1,R2                   ; GIVING END OF MAPPING TABLE ADDRESS
002040  022721  140000                          CMP     #140000,(R1)+           ; IS CHANNEL 0 ASSIGNED?
002044  001435                                  BEQ     30$                     ; IF EQ, NO
002046  005741                                  TST     -(R1)                   ; RESET CHANNEL POINTER
```

```
SF.TER= 001000        S.HAD   000010        T$T5    000030        ZF.INI= 040000        $DEUMR= ****** GX
SF.UNL= 000040        S.HST   000006        T$T6    000032        ZF.KMX= 000020        $DFUIC= ****** GX
SISDR0= 172200        S.ITM   000007        T$$KMG= 000000        ZF.LLC= 000004        $DIGIT= 000024
SLTDF   000776R   003 S.LEN   000004        T$$MIN= 000000        ZF.LMC= 000100        $DNUMB= 000014
SLTDF1  000366R   003 S.LENG  000052        UBMPR = 170200        ZF.MAN= 020000        $EOS  = 000012
SLTDF2  000374R   003 S.LHD   000000        UDSAR0= 177660        ZF.MFL= 000010        $EXIT = 000000
SLTDF3  000350R   003 S.LNK   000000        UDSDR0= 177620        ZF.MTM= 000400        $FAIL = 177777
SLTFLG  001014R   003 S.LOA   000044        UIC     002106R   003 ZF.MUX= 000040        $FILHD= ****** GX
SLTNUM= ****** GX     S.NMST  000002        UICBUF  000206RG  002 ZF.PSE= 002000        $FMASK= ****** GX
SPARE = 000010        S.DWNR  000003        UISAR0= 177640        ZF.SLI= 010000        $GCLEP= ****** GX
SPSAV = ****** GX     S.PKT   000014        UISAR1= 177642        ZF.TIM= 000200        $GETAD= ****** GX
SP.EIP= 000001        S.PRI   000004        UISAR4= 177650        ZF.X3P= 000000        $GETRV= ****** GX
SP.ENB= 000002        S.PSS   000022        UISAR5= 177652        ZS.ASN= 100000        $GTBUF  003132RG
SP.LOG= 000004        S.RCNT  177772        UISAR6= 177654        ZS.BSY= 140000        $HMBUF= ****** GX
SRHFNG  015552R       S.ROFF  177773        UISAR7= 177656        Z.AVL   000014        $LAMDA= 000000
SR0   = 177572        S.SEC   000040        UISDR0= 177600        Z.DAT   000016        $LGSTT= ****** GX
SR3   = 172516        S.STS   000011        UISDR4= 177610        Z.DSP   000000        $LSTAD= ****** GX
STADF   001174R   003 S.TER   000042        UISDR5= 177612        Z.FLG   000010        $LSTRM= ****** GX
STADF1  001214R   003 S.VCT   000005        UISDR6= 177614        Z.LEN = 000016        $MAC  = ****** GX
STAN    000036R   002 TER     002000R   003 UISDR7= 177616        Z.LLN   000006        $MAXOV= ****** GX
STRAD   000332RG  002 TOK     002126R   003 UISZ  = 000014 G      Z.MAP   000020        $MFL    017360RG
STRADD  000056R   002 TOPFLG= ****** GX     UMRBF   000006R   002 Z.NAM   000004        $MUL  = ****** GX
STRADR  010242R       TPS   = 177564        UMRFLG= ****** GX     Z.PCB   000012        $NBIAS= ****** GX
STRING  001736R   003 TYPE    000063R   002 UMRPTR= ****** GX     Z.SCH   000007        $NTCFP= ****** GX
STRMAX  010240R       TY$CHA= 000400        UNT     000061R   002 $ALL15= ****** GX     $NTCFS= ****** GX
STRNG1  001746R   003 TY$EVE= 001000        USRUIC  017142RG      $ALL16= ****** GX     $NTLHB= ****** GX
STRPT   000334RG  002 TY$STA= 002000        USRUI1  017134RG      $ALL22= ****** GX     $NUMBR= 000002
STRPTR  010236R       TY$SUM= 000000        US$CNF= 000002        $ALPHA= 000022        $OBJHD= ****** GX
STRSIZ  000204RG  002 T$FLAG  000044        US$DIS= 000006        $ALPOL= ****** GX     $OPTON= ****** GX
ST$CC = 000004        T$LIF   000013        US$DON= 000000        $ANY  = 000020        $PARHD= ****** GX
ST$CIR= 000006        T$LIFL  000013        US$DSC= 000004        $ASUMR= ****** GX     $PAVL = ****** GX
ST$CIS= 000002        T$LIFO  000013        US$EAC= 000012        $AZFS = ****** GX     $PDVNM= ****** GX
ST$DAT= 000010        T$LIFS  000013        US$WDS= 000010        $BFR  = ****** GX     $PDVTA= ****** GX
ST$DIP= 000012        T$LIN   000000        VF.DLX= ****** GX     $BLANK= 000006        $PUMR = ****** GX
ST$PND= 000014        T$LIPS  000006        VF.NTI= ****** GX     $CAT5 = ****** GX     $PUTAD= ****** GX
SUBPAR= ****** GX     T$LLD   000012        VF.SS = ****** GX     $CBDSG= ****** GX     $PUTRC= ****** GX
SWR   = 177570        T$LLDC  000045        V$$CTR= 001000        $CCBCT= ****** GX     $QBIAS= ****** GX
SYSFDB= ****** GX     T$LLDL  000012        WC.CNT= 000400        $CCBLH= ****** GX     $QSTRT= ****** GX
S$$WRG= 000000        T$LLDO  000012        WC.EVE= 004000        $CCBNM= ****** GX     $QUAL = ****** GX
S$$YSZ= 007600        T$LLDS  000012        WC.TRI= 002000        $CCBS   002554RG      $RADDR= ****** GX
S.ADD   000004        T$LLEN  000046        WC.UNT= 001000        $CCBSZ= ****** GX     $RAD50= 000016
S.BMSK  177776        T$LOPR  000002        XPCIR   004600R       $CEACC= ****** GX     $RBLCK= ****** GX
S.BMSV  177774        T$LTCL  000024        XPTDF1  000262R   003 $CET  = ****** GX     $RDBCT= ****** GX
S.CIR   000016        T$LTIM  000026        XPTLLN  000066R   002 $CFG    005410RG      $RDBLH= ****** GX
S.CON   000010        T$LTPR  000014        X$$DBT= 000000        $CLBUF= ****** GX     $RDBNM= ****** GX
S.COST  000001        T$LTPS  000020        X.FLG   010244R       $CLDEQ= ****** GX     $RDBS   002644RG
S.CSR   000012        T$NAPL  000004        X.FLG1  010336R       $CLFLG= ****** GX     $RDBSZ= ****** GX
S.CTM   000006        T$NFE   000000        X1      000144R   003 $CLIOS= ****** GX     $RDBTH= ****** GX
S.DEV   000020        T$NLEN  000010        X2      000156R   003 $CLQUE= ****** GX     $RET1   001356RG
S.DIA   000046        T$NNUL  000002        X3      000210R   003 $CLSAV= ****** GX     $RET2   001400RG
S.DMCS  000020        T$NOPL  000006        X4      000224R   003 $CMAND= ****** GX     $RSIZE= ****** GX
S.DPA   000032        T$NRNI  000042        ZAS     002542R   003 $CRPAR  003366RG      $RSNTI  020156RG
S.DPC   000036        T$NRPL  000005        ZF.COU= 001000        $CTIM = ****** GX     $SAVAL= ****** GX
S.DUM   000050        T$NRUL  000007        ZF.DDM= 000001        $CXOPT= ****** GX     $SAVRG= ****** GX
S.FLAG  000002        T$NVR   000001        ZF.DIA= 004000        $DALCT  003200RG      $SAVWR= ****** GX
S.FLG   000000        T$RPRI  000040        ZF.DLC= 000002        $DEA16= ****** GX     $SDBCT= ****** GX
S.FRK   000016        T$SVC   000034        ZF.DVP= 100000        $DECPT= ****** GX     $SDBLH= ****** GX
```

```
                                                                   C  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50     PAGE 12
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE              REFERENCES

PBLOX    = ******  GX       *8-304      8-377     11-690     11-697     26-1552   *26-1560   *26-1625   *26-1636
PDVNAM   = ******  GX       *55-2625    56-2642   59-2695
PIRQ     = 177772           #5-75
PMODE    = 030000           #5-75
PREFIX     001734 R          9-542     #10-612    10-613
PRELEN   = 000016            9-543     #10-613
PREWLN   = 0000 4            10-585    #10-615
PREWRN     001  2 R          10-584    #10-614    10-615
PROTXT     006 56 R          28-1705   #31-1866
PRVUNM     000002 R         #6-115     *80-3538   80-3545   *87-3722    87-3729   *88-3754    88-3765
PR0      = 000000           #5-75
PR1      = 000040           #5-75
PR2      = 000100           #5-75
PR3      = 000140           #5-75
PR4      = 000200           #5-75
PR5      = 000240           #5-75
PR6      = 000300           #5-75
PR7      = 000340           #5-75
PS       = 177776           #5-75
PSTART   = ******  GX       *8-302      8-367     8-376      11-709     26-1553   *26-1585   *26-1591   *26-1609    26-1623
                            *26-1624   *26-1635
PS.COM   = 000200            8-275      26-1566
PS.NSF   = 000400            26-1566
PS.SYS   = 000040            17-1057    26-1566
PUMR     = ******  GX       *8-424      8-427    *11-740
P.BLKS     000016           *26-1559    26-1595   26-1599
P.BUSY     000024            8-278      8-289     8-291     *8-291      8-297     *8-297
P.MAIN     000012            8-281     *26-1558
P.NAM      000004            8-271      8-273     17-1063    17-1065    17-1070    17-1072   *26-1549   *26-1550   *26-1556
                            *26-1557
P.REL      000014            8-301      17-1081   26-1574    26-1578   *26-1584   *26-1590    26-1598   *26-1608
P.SIZE     0000              8-303      17-1082
P.STAT     000030            8-275      17-1057  *26-1566
P.SUB      000010            26-1575   *26-1586  *26-1610   *26-1615
QF$ACT   = 000001           #5-78
QF$ALL   = 000002           #5-78
QF$KNO   = 000004           #5-78
QF$LOO   = 000010           #5-78
QINS       017272 R          69-3188    79-3514   86-3695    94-4073    102-4464  #105-4700
QIO        002006 R         *9-542     *9-543    *9-544      9-545     *9-551     *9-552     *9-553      9-554     *9-557
                            *9-558     *9-559     9-560     *10-584    *10-585    *10-586     10-587    *10-592    *10-593
                            *10-594     10-595   *10-603    *10-604    *10-605     10-606    #10-624
QUICK      014102 R          69-3178   #89-3788
Q.IOFN   = 000002            10-624
Q.IOPL   = 000014           *9-542     *9-543    *9-544     *9-551     *9-552     *9-553     *9-557     *9-558     *9-559
                            *10-584    *10-585   *10-586    *10-592    *10-593    *10-594    *10-603    *10-604    *10-605
RAD        000104 R         #6-142      24-1383  *25-1508
RBRA     = 000076           #27-1667
REMARE     000100 R         #6-140      69-3170  *69-3214
REQNTL     020140 R          8-368      16-964    107-4789   108-4827   109-4861  #110-4877
RETF       011146 R         #67-3101
RF.100   = 100000           #5-64
```

```
                                                                   D  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50     PAGE 13
```

```
                                                                      C  3
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 25

MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME        REFERENCES

                  #49-2486       #49-2490       #49-2495       #49-2500       #49-2503       #49-2508       #49-2512       #49-2515       #49-2518       #49-2521
                  #49-2524
SWTCHI            #5-57           8-286          8-288          61-2752        61-2754        64-2888        64-2890        65-2919        65-2922        69-3183
                   69-3185        88-3746        88-3748        94-4062        94-4064        98-4257        98-4259        102-4453       102-4455
SWTCHO            #5-57           8-292          8-294          8-373          1  741         11-743         61-2777        64-2892        64-2895        65-2923
                   65-2959        65-2961        69-3190        69-3192        94-4075        94-4077        102-4466       102-4468
TRAN$             #5-56          #18-1111       #18-1113       #18-1115       #18-1117       #18-1119       #18-1121       #18-1123       #18-1125       #18-1127
                  #18-1129       #18-1131       #18-1133       #19-1142       #19-1144       #19-1146       #19-1148       #19-1150       #19-1152       #19-1154
                  #19-1156       #19-1158       #19-1160       #19-1162       #19-1164       #19-1166       #19-1168       #19-1170       #19-1172       #20-1181
                  #20-1183       #20-1184       #20-1186       #20-1188       #20-1189       #20-1191       #20-1193       #20-1195       #20-1196       #20-1198
                  #20-1199       #20-1201       #20-1202       #20-1204       #20-1205       #20-1207       #20-1208       #20-1210       #20-1212       #20-1213
                  #20-1215       #21-1223       #21-1226       #21-1229       #21-1232       #21-1233       #21-1234       #21-1237       #21-1240       #21-1243
                  #22-1253       #22-1255       #22-1257       #22-1259       #23-1268       #23-1270       #23-1272       #23-1274       #23-1276       #23-1278
                  #23-1280       #23-1282       #23-1283       #23-1285       #23-1286       #23-1287       #23-1288       #23-1290       #23-1291       #23-1293
                  #23-1295       #23-1297       #23-1299       #23-1300       #23-1302       #23-1303       #23-1305       #23-1306       #23-1308       #23-1309
                  #23-1311       #23-1312       #23-1314       #23-1315       #23-1317       #23-1318       #23-1320       #32-1891       #32-1892       #32-1893
                  #32-1894       #32-1895       #32-1896       #32-1897       #32-1898       #32-1899       #32-1900       #32-1901       #32-1902       #32-1903
                  #32-1904       #32-1905       #32-1906       #32-1907       #32-1908       #33-1917       #33-1918       #33-1920       #33-1921       #34-1930
                  #34-1932       #34-1934       #34-1935       #34-1936       #34-1937       #34-1938       #34-1940       #34-1942       #34-1943       #34-1945
                  #34-1947       #34-1948       #35-1957       #35-1959       #35-1961       #35-1963       #35-1965       #35-1966       #35-1967       #35-1968
                  #35-1969       #35-1970       #35-1972       #35-1974       #35-1976       #35-1978       #35-1979       #35-1981       #35-1982       #35-1984
                  #35-1985       #35-1987       #35-1988       #35-1990       #35-1991       #35-1993       #35-1994       #35-1995       #35-1997       #35-1998
                  #35-2000       #36-2010       #36-2012       #36-2014       #36-2016       #36-2017       #36-2019       #36-2020       #36-2021       #36-2022
                  #36-2023       #36-2024       #37-2031       #38-2039       #38-2041       #38-2043       #38-2045       #38-2047       #38-2049       #38-2051
                  #38-2053       #38-2055       #38-2057       #38-2059       #38-2061       #38-2063       #39-2071       #39-2074       #39-2077       #39-2078
                  #39-2081       #39-2084       #39-2087       #39-2090       #39-2093       #39-2096       #39-2099       #39-2100       #39-2103       #40-2113
                  #40-2116       #40-2119       #40-2122       #40-2125       #40-2128       #40-2131       #40-2134       #40-2137       #40-2140       #40-2143
                  #40-2146       #40-2149       #40-2152       #40-2155       #40-2158       #40-2161       #40-2164       #40-2167       #40-2170       #40-2173
                  #41-2185       #41-2188       #41-2191       #41-2194       #41-2197       #41-2200       #41-2203       #42-2210       #42-2213       #42-2216
                  #42-2219       #42-2222       #42-2225       #42-2226       #42-2227       #42-2233       #42-2236       #42-2237       #43-2244       #44-2252
                  #44-2254       #44-2256       #44-2258       #44-2260       #44-2262       #44-2264       #44-2266       #44-2268       #44-2270       #44-2272
                  #44-2274       #44-2276       #44-2278       #44-2280       #44-2282       #44-2284       #45-2295       #45-2296       #45-2303       #45-2305
                  #45-2313       #45-2314       #45-2316       #46-2333       #46-2334       #46-2335       #46-2338       #46-2339       #46-2342       #46-2343
                  #46-2346       #46-2347       #46-2350       #46-2351       #46-2354       #46-2355       #46-2358       #46-2359       #46-2362       #46-2363
                  #46-2366       #46-2369       #47-2374       #47-2375       #47-2376       #47-2379       #47-2382       #47-2383       #47-2385       #47-2386
                  #47-2387       #47-2388       #47-2391       #47-2394       #48-2402       #48-2405       #48-2406       #48-2407       #48-2410       #48-2413
                  #48-2414       #48-2417       #48-2418       #48-2421       #48-2422       #48-2425       #48-2428       #48-2429       #48-2432       #48-2433
                  #48-2436       #48-2437       #48-2440       #48-2443       #48-2446       #48-2447       #48-2450       #48-2451       #48-2454       #48-2455
                  #48-2458       #48-2459       #49-2467       #49-2468       #49-2471       #49-2472       #49-2475       #49-2476       #49-2479       #49-2480
                  #49-2483       #49-2484       #49-2487       #49-2488       #49-2491       #49-2492       #49-2496       #49-2497       #49-2501       #49-2504
                  #49-2509       #49-2510       #49-2513       #49-2516       #49-2519       #49-2522
VNPBPT            #8-217          8-219         #110-4882       110-4883
VNPDF$            #5-55           5-78
```

```
                                                                      D  3
**FILE**ID**VPIO
```

```

                                               ;+
                                               ; SFIL - SET UP MICROCODE TASK FILESPEC
                                               ;
                                               ; INPUTS:
                                               ;        $DFUIC - DEFAULT DEV:[UIC]
                                               ;
                                               ; OUTPUTS:
                                               ;        FILE SPEC SET UP IN MCFBF BUFFER
                                               ;
                                               ;-

000624  005002                                 SFIL:   CLR     R2              ; FILESPEC LENGTH := 0 ;
000626  012703  000212'                                MOV     #MCFBF,R3       ; STORE BUFFER ADDRESS
000632  012700  000000G                                MOV     #$DFUIC,R0      ; STORE DEFAULT UIC ADDRESS

000636  005202                                 10$:    INC     R2              ; BUMP LENGTH
000640  112023                                         MOVB    (R0)+,(R3)+     ; COPY BYTE
000642  001375                                         BNE     10$             ; IF MORE - BRANCH
000644  005302                                         DEC     R2              ; ADJUST LENGTH
000646  005303                                         DEC     R3              ; ...

000650  022767  102561  000000G                        CMP     #^RUNA,$NAME    ; UNA DEVICE ?
000656  001003                                         BNE     15$             ; IF NO - BRANCH
000660  012700  000154'                                MOV     #UNAMC,R0       ; SET FILE NAME
000664  000411                                         BR      40$             ; CONTINUE
000666  022767  042560  000000G        15$:    CMP     #^RKDP,$NAME    ; KDP DEVICE ?
000674  001003                                         BNE     20$             ; IF NO - BRANCH
000676  012700  000166'                                MOV     #KDPMC,R0       ; SET FILE NAME
000702  000402                                         BR      40$
000704  012700  000200'                        20$:    MOV     #KDZMC,R0       ; SET FILE NAME

000710  005202                                 40$:    INC     R2              ; BUMP LENGTH
000712  112023                                         MOVB    (R0)+,(R3)+     ; COPY BYTE
000714  001375                                         BNE     40$             ; IF MORE - BRANCH
000716  005302                                         DEC     R2              ; ADJUST LENGTH
000720  012701  000212'                                MOV     #MCFBF,R1       ; STORE BUFFER ADDRESS

000724                                                 RETURN



                                               ;+
                                               ; RMCLBL - READ LABEL BLOCK
                                               ;
                                               ; INPUTS:
                                               ;        FILE OPEN
                                               ;
                                               ; OUTPUTS:
                                               ;        FILE SIZE SET UP FOR TRANSFER
                                               ;+

000726  012700  000000'                        RMCLBL: MOV     #DSKBUF,R0      ; STORE BUFFER ADDRESS
000732                                                 CALL    $RLBL           ; READ LABEL BLOCK
000736  103420                                         BCS     10$             ; IF ERROR - BRANCH
000740  016001  000000G                                MOV     L$BLDZ(R0),R1   ; COPY SIZE
000744  010167  000260'                                MOV     R1,$LDSIZ       ; ...
000750  010167  000142'                                MOV     R1,BLOX         ; ...
000754                                                 ASL$    6,R1            ; CONVERT TO BYTES
```

```
Symbol table

AS.DEL= 000010          D$NN    000060          FE.EXT= 000001          HF.QB = 000004          N$DLA   000020
AS.EXT= 000004          D$OUTT  000043          FE.EXV= 000004          HF.UBM= 000001          N$DLY   000014
AS.RED= 000001          D$RETF  000050          FE.FDT= 002000          I$$RAR= 000000          N$ELEN  000054
AS.WRT= 000002          D$RNN   000002          FE.LSI= 000400          I$$RDN= 000000          N$ENC   000042
A$$CHK= 000000          D$RTMR  000076          FE.MUP= 000002          KDPMC   000166R     002 N$ERRC  000022
A$$CPS= 000000          D$SEG   000036          FE.MXT= 040000          KDSAR0= 172360          N$FLG   000005
A$$PRI= 000000          D$SER   000032          FE.NLG= 100000          KDSDR0= 172320          N$FNC   000006
A$$TRP= 000000          D$SQRL  000052          FE.OFF= 001000          KDZMC   000200R     002 N$GENQ  000052
A.IOC   000003          D$$BUG= 177514          FE.PKT= 000100          KINAR0= 172340          N$GTM   000015
A.LGTH= 000014          D$$ISK= 000000          FE.PLA= 000020          KINAR5= 172352          N$HIGH  000033
A.MPCT  000011          D$$L11= 000001          FE.X25= 004000          KINAR6= 172354          N$LLT   000026
A.PCB   000012          D$$YNC= 000000          FL.CEX= ****** GX       KINAR7= 172356          N$LLTM  000024
A.PCBL  000000          D$$YNM= 000000          FL.LLC= ****** GX       KISAR0= 172340          N$LVC   000036
A.PRI   000002          ERROR   000250R     002 FL.LMC= ****** GX       KISAR5= 172352          N$MBXQ  000050
A.STAT  000010          ERR10   000012          FL.TOP= ****** GX       KISAR6= 172354          N$PLLT  000030
A.TCB   000004          ERR12   000014          FNDPDV  003100R         KISAR7= 172356          N$SLA   000016
A.TCBL  000006          ERR14   000016          F$$LVL= 000001          KISDR0= 172300          N$SNOD  000012
BLOX    000142R     002 ERR16   000020          F.HIBK= ****** GX       KISDR6= 172314          N$TIM   000004
BYTES   000144R     002 ERR18   000022          F.RSIZ= ****** GX       KISDR7= 172316          N$VCB   000010
CMODE = 140000          ERR2    000002          F2.ACN= 000020          K$$CNT= 177546          N$$ACC= 000001
CR.MC   000220R         ERR20   000024          F2.AHR= 010000          K$$CSR= 177546          N$$BUF= 000001
C$LEN   000012          ERR22   000026          F2.DAS= 000001          K$$LDC= 000000          N$$LDV= 000001
C$NID   000000          ERR24   000030          F2.DPR= 000400          K$$TPS= 000074          N$$MCP= 000001
C$NXH   000002          ERR26   000032          F2.EVT= 000010          LD$LP = 000000          N$$MLL= 000001
C$TIM   000010          ERR28   000034          F2.GGF= 002000          L$BEXT= ****** GX       N$$MOV= 000010
C$$CKP= 000000          ERR30   000036          F2.IRR= 001000          L$BFLG= ****** GX       N$$NCT= 000001
C$$ORE= 000400          ERR32   000040          F2.LIB= 000002          L$BLDZ= ****** GX       N$$PEM= 000001
C$$RSH= 177564          ERR34   000042          F2.MP = 000004          L$BMXZ= ****** GX       PCBA    000252R     002
DSKBUF  000000R     003 ERR4    000004          F2.POL= 000100          L$BPAR= ****** GX       PIRQ  = 177772
D$AMXC  000072          ERR6    000006          F2.RAS= 004000          L$BSA = ****** GX       PMODE = 030000
D$AMXH  000074          ERR8    000010          F2.RBN= 020000          L$$ASG= 000000          PR.LBL  001002R
D$ANN   000000          EXTEND  000146R     002 F2.SDW= 000040          L$$DRV= 000000          PR.PAR  001540R
D$BRPR  000102          EXTXPT  001424R         F2.STP= 100000          L$$P11= 000001          PR.SUB  002334R
D$BRTM  000100          E$NBR   000014          F2.SWP= 040000          L$$11R= 000000          PR.XFR  003026R
D$DELF  000045          E$NBS   000020          F2.WND= 000200          MBUF    000262R     002 PR0   = 000000
D$DELW  000046          E$NCR   000034          F3.AST= 000200          MCFBF   000212R     002 PR1   = 000040
D$END = 000104          E$NCS   000036          F3.CLI= 001000          MCFLG   000150R     002 PR4   = 000200
D$FNB   000034          E$NIC   000044          F3.CRA= 000001          MCLOA = 000001          PR5   = 000240
D$HIOR  000024          E$NLEN  000050          F3.EIS= 000004          MPAR  = 172100          PR6   = 000300
D$HOST  000022          E$NLLA  000012          F3.NWK= 000002          MPCSR = 177746          PR7   = 000340
D$INAC  000044          E$NLNK  000000          F3.PMN= 004000          MSGPRI  003152R         PS    = 177776
D$INCT  000042          E$NML   000040          F3.PRO= 000040          M$$CRB= 000124          PS.APR= 000007
D$IPL   000051          E$NMR   000024          F3.RLK= 020000          M$$CRX= 000000          PS.CHK= 010000
D$LID   000020          E$NMS   000030          F3.SHF= 040000          M$$FCS= 000000          PS.CKP= 040000
D$LNAM  000006          E$NNOD  000002          F3.STM= 000010          M$$MGE= 000000          PS.CKR= 020000
D$LNUM  000014          E$NRT   000042          F3.TCM= 002000          M$$NET= 000000          PS.COM= 000200
D$LST   000047          E$NRTP  000005          F3.UDS= 000020          M$$OVR= 000000          PS.DEL= 000010
D$MAXC  000064          E$NSEG  000010          F3.WAT= 010000          NF$BLK= 000100          PS.DRV= 000020
D$MAXH  000066          E$NTIM  000046          F3.XHR= 000100          NF$DMO= 000010          PS.FXD= 004000
D$MAXV  000070          E$NUSE  000004          F3.11S= 000400          NF$MOU= 000040          PS.LIO= 001000
D$MLL   000040          E$STRT  000006          G$$TPP= 000000          NF$RST= 000002          PS.NSF= 000400
D$MNOD  000041          E$$XPR= 000000          G$$TSS= 000000          NF$SCN= 000020          PS.OUT= 100000
D$NA    000062          FE.CAL= 000040          G$$TTK= 000000          NF$SHU= 000004          PS.PER= 002000
D$NBEA  000056          FE.CEX= 020000          G$$WRD= 000000          NF$TIM= 000200          PS.PIC= 000100
D$NBRA  000054          FE.DRV= 000010          HF.CIS= 000200          N$ACQ   000000          PS.SYS= 000040
D$NEND= 000054          FE.DYM= 010000          HF.EIS= 000002          N$ACTL  000032          P$$P45= 000000
D$NLN   000030          FE.EXP= 000200          HF.FPP= 100000          N$CIR   000034          P$$WRD= 000000
```

```
    132 000000                                    PRMPT$  <'Process name: '>,                P$PID,<:>
    133 000000                                    PRMPT$  <'Scope: '>,                       P$SCO,<:>
    134 000000                                    PRMPT$  <'Set component: '>,               P$SET,<:>
    135 000000                                    PRMPT$  <'Set alias options: '>,           P$SEAL,<:>
    136 000000                                    PRMPT$  <'Set circuit options: '>,         P$SECI,<:>
    137 000000                                    PRMPT$  <'Set line options: '>,            P$SELI,<:>
    138 000000                                    PRMPT$  <'Set logging options: '>,         P$SELO,<:>
    139 000000                                    PRMPT$  <'Set node options: '>,            P$SENO,<:>
    140 000000                                    PRMPT$  <'Set module options: '>,          P$SEMO,<:>
    141 000000                                    PRMPT$  <'Set object options: '>,          P$SEOB,<:>
    142 000000                                    PRMPT$  <'Set process options: '>,         P$SEPR,<:>
    143 000000                                    PRMPT$  <'Show component: '>,              P$SHO,<:>
    144 000000                                    PRMPT$  <'Sink node identification: '>,    P$SIN,<:>
    145 000000                                    PRMPT$  <'Sink name: '>,                   P$SNAM,<:>
    146 000000                                    PRMPT$  <'State: '>,                       P$STA,<:>
    147 000000                                    PRMPT$  <'Terminal identification: '>,     P$TER,<:>
    148 000000                                    PRMPT$  <'Text password: '>,               P$TPAS,<:>
    149 000000                                    PRMPT$  <'Tributary address: '>,           P$TRIB,<:>
    150 000000                                    PRMPT$  <'User identification: '>,         P$USE,<:>
    151 000000                                    PRMPT$  <'Vector: '>,                      P$VEC,<:>
    152 000000                                    PRMPT$  <'Verification state: '>,          P$VEST,<:>
    153 000000                                    PRMPT$  <'X25-server state: '>,            P$X5ST,<:>
    154
    155             000001                        .END
    326                                                 .SBTTL  ALVBLK - ALLOCATE VECTOR BLOCK
    327                                         ;+
    328                                         ; ALVBLK - ALLOCATE VECTOR BLOCK
    329                                         ;
    330                                         ; INPUTS:
    331                                         ;       VECVAL - ADDRESS OF VECTOR TABLE
    332                                         ;       VLNVAL - SIZE OF VECTOR TABLE (WORDS)
    333                                         ;
    334                                         ; OUTPUTS:
    335                                         ;       VECTOR BLOCK ALLOCATED AND INITIALIZED
    336                                         ;-
    337 000346                                  ALVBLK: SAVRG   <R1>                            ; SAVE PCB ADDRESS
    338 000350  012701  000012                          MOV     #V.LEN,R1                       ; GET SIZE OF VECTOR BLOCK
    339 000354                                          CALL    $ALPOL                          ; ALLOCATE FROM BYTE POOL
    340 000360  016767  000026G 000000G                 MOV     $NTLHB+.CXVEC,$BFR+V.LNK ; GET ADDRESS OF OLD FIRST BLOCK
    341 000366  010067  000026G                         MOV     R0,$NTLHB+.CXVEC                ; LINK IN NEW VECTOR BLOCK
    342
    343 000372  005067  000002G                         CLR     $BFR+V.FLG                      ; CLEAR VECTOR BLOCK FLAGS WORD
    344 000376  016767  000054' 000006G                 MOV     VECVAL,$BFR+V.ADD               ; COPY TABLE ADDRESS
    345 000404  016767  000066' 000010G                 MOV     VLNVAL,$BFR+V.SIZ               ; COPY TABLE SIZE
    346 000412  016746  000000G                         MOV     $PDVA,-(SP)                     ; COPY INTERNAL PDV ADDRESS
    347 000416  066716  000000G                         ADD     $CETDF,(SP)                     ; CALCULATE ACTUAL PDV ADDRESS
    348 000422  012667  000004G                         MOV     (SP)+,$BFR+V.PDV                ; STORE PDV ADDRESS
    349
    350 000426                                          CEACC$  R0,R2                           ; MAP TO VECTOR BLOCK
    351 000436                                          PUTAD   R2,#V.LEN                       ; UPDATE IMAGE
    352 000454                                          RESRG   <R1>                            ; RESTORE PCB ADDRESS
    353
    354 000456                                          RETURN
    355
    968
```

```
                                                                   C  9
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05        PAGE 4

SYMBOL CROSS REFERENCE                                      CREF   04.00

SYMBOL   VALUE               REFERENCES

R$$11M  = 000000               12-647      15-947      16-1018     17-1117     17-1127     17-1140
SECBLK    000152 R              6-181     #6-185        9-398
SF.ACT  = 000200               #5-72
SF.ENA  = 000100               #5-72
SF.LPB  = 000004               #5-72
SF.MFL  = 000040               #5-72
SF.PAC  = 000020               #5-72
SF.REA  = 000010               #5-72
SF.SER  = 000001               #5-72
SF.SVC  = 000002               #5-72
SF.UNL  = 000040               #5-72
SQBLK     000126 R              6-170     #6-177        9-390
STBERR    001054 R             10-480    #10-506
STBEXT    000020 R             #6-117     10-476
SYSFDB  = ******   GX          *7-270     *7-308       *7-311      *8-351      *11-571     *11-577     *11-590     *12-618     *12-640
                               *12-681    *13-707      *13-720     *13-750     *13-774     *13-788     *14-817     *14-835     *14-838
                               *14-845    *15-877      *15-899     *15-917     *15-928     *15-942     *15-951     *16-989     *16-1010
                               *16-1053   *16-1065     *17-1096    *17-1111    *17-1123    *17-1141
S.COST    000001               #5-72
S.FLG     000000               #5-72
S.LEN     000004               #5-72
S.NMST    000002               #5-72
S.OWNR    000003               #5-72
TSKERR    000340 R              7-260     #7-322
T$FLAG    000044               #5-68
T$LIF     000013               #5-68
T$LIFL    000013               #5-68
T$LIFO    000013               #5-68
T$LIFS    000013               #5-68
T$LIN     000000               #5-68
T$LIPS    000006               #5-68
T$LLD     000012               #5-68
T$LLDC    000045               #5-68
T$LLDL    000012               #5-68
T$LLDO    000012               #5-68
T$LLDS    000012               #5-68
T$LLEN    000046               #5-68       11-577      11-582      11-590      15-932      15-951      15-954      15-959
T$LOPR    000002               #5-68
T$LTCL    000024               #5-68
T$LTIM    000026               #5-68
T$LTPR    000014               #5-68
T$LTPS    000020               #5-68
T$NAPL    000004               #5-68
T$NFE     000000               #5-68
T$NLEN    000010               #5-68
T$NNUL    000002               #5-68
T$NOPL    000006               #5-68
T$NRNI    000042               #5-68      *11-588
T$NRPL    000005               #5-68
T$NRUL    000007               #5-68
T$NVR     000001               #5-68
T$RPRI    000040               #5-68      *11-589
```

```
                                                                   D  9
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05        PAGE 5
```

```
    147                                         ;
    148                                         ; NETWORK INITIALIZATION OVERLAY RESIDENT DATA
    149                                         ;
    150
    151                                         .IF NDF R$$MPL                                                  ;[MP01]
    152         000044                          DSIZE  == 36.              ; FIXED PACKET LENGTH                ;[MP01]
    153
    154                                         .IFF
    155                                         $SVCOL == 10               ; LOACTION FOR VBN OF DISPLACED MEMORY  ;[MP01]
    156                                         DSIZE  == 13.+8.+37.       ; VARIABLE LENGTH PACKET             ;[MP01]
    157                                         $URM::   .WORD   0         ; UNIBUS RUN MASK                    ;[MP01]
    158                                         $RCDEV::.WORD    0         ; RECONFIGURATION DEVICE             ;[MP01]
    159                                         $CTBIM::.WORD    0         ; ACTUAL CTB ADDRESS                 ;[MP01]
    160                                         $CTBAD::.WORD    0         ; CTB ADDRESS IN CETAB               ;[MP01]
    161                                         CTBSAV::.WORD    0         ; SAVE CTB ADDRESS IF SAME DEVICE TYPE  ;[MP01]
    162                                         $LIBNM::.BLKB    10.       ; ASCII NAME OF LIBRARY              ;[MP01]
    163                                         $LBVL::  .BLKW   1         ; VALUE OF $XXXVL FOR LIBRARY        ;[MP01]
    164                                         $LBVT::  .BLKW   1         ; VALUE OF $XXXVT FOR LIBRARY        ;[MP01]
    165                                         $LBBLK::.BLKW    1         ; VALUE OF LIBRARY BIAS              ;[MP01]
    166
    167                                         .ENDC                                                           ;[MP01]
    168
    169 000722  000000                          $LSTRM::.WORD    0         ; LAST REMOTE BLOCK UNMAPPED ADDRESS
    170 000724  000000                          $LSTAD::.WORD    0         ; HIGHEST NODE ADDRESS
    171 000726  000000                          $BIGCT::.WORD    0         ; FLAG FOR CETAB > 2000 BYTES
    172 000730  000000                          RA::     .WORD   0         ; VIRTUAL ADDRESS FOR D-DPACE I/O    ;[MP01]
    173 000732  000000                          RB::     .WORD   0         ; SAVED BIAS BEFORE D-SPACE          ;[MP01]
    174 000734  000000  000000                  NTPNAM::.WORD    0,0       ; NETWORK POOL NAME IN RAD50
    175 000740  000000  000000                  NTPPAR::.WORD    0,0       ; PARTITION NAME TO ALLOCATE SUB-PAR IN
    176 000744  000000                          SUBPAR::.WORD    0         ; POOL.. SUBPARTITION PCB ADDRESS
    177 000746  000000                          SBPCB::  .WORD   0         ; SERVER BASE BUFFER POOL PCB ADDRESS
    178 000750  000000  000000                  NTPSIZ::.WORD    0,0       ; SUBPARTITION SIZE IN MAPPED SYSTEMS
    179 000754  000000                          TOPFLG::.WORD    0         ; NON-ZERO CAUSES TOP-DOWN LOADING
    180 000756  000000                          NTPSAV::.WORD    0         ; POOL.. SIZE TEMP LOCATION
    181 000760  000000                          SBPOOL::.WORD    0         ; SERVER BASE BUFFER FLAG
    182 000762  055260                          SBPAR::  .WORD   ^RNTP     ; SERVER BASE MAIN PARTITION NAME
    183 000764  060044                                   .WORD   ^ROOL     ; ...
    184 000766  073440                          SBNAM::  .WORD   ^RSBP     ; SERVER BASE SUB-PARTITION BUFFER NAME
    185 000770  060044                                   .WORD   ^ROOL     ; ...
    186 000772  000000                          SBFLG::  .WORD   0         ; SERVER BASE TPARS FLAG
    187 000774                                  UMRPTR::.BLKW    1         ; UMR BLOCK REAL MEMORY ADDRESS
    188 000776                                  UMRFLG::.BLKW    1         ; NON-ZERO DISABLES UMR ALLOCATION
    189 001000                                  NONMAP::.BLKW    1         ; SIZE OF NON-UMR-MAPPED AREA
    190 001002                                  PSTART::.BLKW    1         ; NETWORK POOL BIAS
    191 001004                                  PBLOX::  .BLKW   1         ; NETWORK POOL SIZE IN 32. WORD BLOCKS
    192 001006                                  SPSAV::  .BLKW   1         ; SAVED SP ON ENTRY AT $VNTI
    193 001010                                  PUMR::   .BLKW   1         ; USED TO CALCULATE $PUMR VALUE
    194 001012                                  .ALLOC::.BLKW    1         ; SIZE OF NETWORK BYTE AREA
    195 001014                                  .RDBNM::.BLKW    1         ; NUMBER OF RDB'S
    196 001016                                  .RDBSZ::.BLKW    1         ; RDB SIZE
    197 001020                                  .SDBNM::.BLKW    1         ; NUMBER OF SDB'S
    198 001022                                  .SDBSZ::.BLKW    1         ; SDB SIZE
    199
    200
    201 000000                                          .PSECT  TMP$$A,RW,OVR   ; TEMPLATE PROCESSING BUFFER
    202
    203 000000                                  $TMPBF::                   ; TEMPLATE PROCESSING BUFFER
    644                                                .SBTTL  RSX DATA FROM RSX11M.STB
    645                                        ;
    646                                        ; VNP "IN CORE" IMAGES OF DISK FILES
    647                                        ;
    648                                        ;
    649                                        ; VALUES LOOKED UP IN RSX11M.STB ( -1 = UNDEFINED )
    650                                        ;
    651                                        ;       NOTE - SYMBOLS IN THIS TABLE MUST BE IN SAME ORDER AS SYMBOLS
    652                                        ;       IN "VINI".
    653                                        ;
    654
    655 003550                                 $SYMTB::                                ; REFERENCE POINT
    656
    657                                                .IF NDF R$$MPL
    658 003550  000000                         S.MPR:  .WORD   0                       ; USED ONLY TO FORCE ERROR IF ON M+     ;[MP01]
    659                                                .IFF
    660                                        .MPRO:: .WORD   0                       ; NUMBER OF PROCESSORS OR -1 IF ONE     ;[MP01]
    661                                        .KDAS:: .WORD   0                       ; KERNAL DATA SPACE ENABLED OR -1 IF NOT
    662                                        .SCMOF::.WORD   0                       ; ADDRESS OF 4K DATA SPACE BIAS
    663                                        .CPPAR::.WORD   0                       ; POINTER TO PCB OF CPU PARTITION
    664                                        .K6TAB::.WORD   0                       ; TABLE OF POINTERS TO EACH PROCESSOR'S AREA
    665                                        .M100Q::.WORD   0                       ; 100 MSEC TIMER OFFSET INTO CPU AREA   ;[MP01]
    666                                        .CPURM::.WORD   0                       ; CPU URM MAPPING TABLE POINTER
    667                                        .UMRPT::.WORD   0                       ; PER CPU NETWORK POOL UMR POINTER
    668                                        .URMTB::.WORD   0                       ; UNIBUS RUN CONTINUITY TABLE
    669                                        .STMFC::.WORD   0                       ; 100 MSEC TIMEOUT FUNCTION CODE
    670                                        .LTMFC::.WORD   0                       ; LONG TIMEOUT FUNCTION CODE
    671                                        .CCBAF::.WORD   0                       ; ALLOCATION FAILURES
    672                                        .RDBAF::.WORD   0                       ; ...
    673                                        .LDBAF::.WORD   0                       ; ...
    674                                        .SDBAF::.WORD   0                       ; ...
    675                                        .CCBAL::.WORD   0                       ; NUMBER OF DYNAMIC CCB ALLOCATIONS
    676                                        .CMFRK::.WORD   0                       ; FORK BLOCK LISTHEAD
    677                                        .CEX::  .WORD   0                       ; ADDRESS OF COM EXEX IN RSX EXEC SPACE
    678                                        .OBJHD::.WORD   0                       ; OBJECT BLOCK LISTHEAD                 ;[MP01]
    679                                        .FILHD::.WORD   0                       ; FILTER BLOCK LISTHEAD                 ;[MP01]
    680                                        .CCBLH::.WORD   0                       ; CCB LISTHEAD                          ;[MP01]
    681                                        .SDBLH::.WORD   0                       ; SDB LISTHEAD                          ;[MP01]
    682                                        .RDBLH::.WORD   0                       ; RDB LISTHEAD                          ;[MP01]
    683                                        .SKS5:: .WORD   0                       ; S.KS5 SCB OFFSET                      ;[MP01]
    684                                        .NTLPT::.WORD   0                       ; NTL HOME BLOCK                        ;[MP01]
    685                                        .PSIPT::.WORD   0                       ; PSI HOME BLOCK POINTER                ;[MP01]
    686                                        .POLHD::.WORD   0                       ; SECONDARY POOL LISTHEAD               ;[MP01]
    687                                        .SECFR::.WORD   0                       ; SECONDARY POOL BLOCKS FREE            ;[MP01]
    688                                        .PDVTA::.WORD   0                       ; PDV TABLE                             ;[MP01]
    689                                        .PUMRP::.WORD   0                       ; PUMR ADDRESS                          ;[MP01]
    690                                        .CTLST::.WORD   0                       ; CTB LIST HEAD                         ;[MP01]
    691                                        .ICAVL::.WORD   0                       ; ICB AVAILABLE LIST                    ;[MP01]
    692                                                .ENDC
    693
    694 003552  000000                         .CRAVL::.WORD   0                       ; DYNAMIC STORAGE REGION LISTHEAD
    695 003554  000000                         ..COO:: .WORD   0                       ; ADDRESS CONSOLE UOUTPUT UCB
    696 003556  000000                         .DEVHD::.WORD   0                       ; DEVICE TABLE LISTHEAD
    697 003560  000000                         .EXSIZ::.WORD   0                       ; HIGHEST ADDRESS IN EXEC
    698 003562  000000                         .FMASK::.WORD   0                       ; SYSTEM FEATURE MASK
    699 003564  000000                         .PARHD::.WORD   0                       ; PARTITION CONTROL BLOCK LISTHEAD
    700 003566  000000                         .SYSIZ::.WORD   0                       ; SYSTEM SIZE IN 32-WORD BLOCKS
```

```
                                                 .SBTTL  $VGCL - GET COMMAND LINE
                                         ;+
                                         ;
                                         ;       **-$VGCL
                                         ;
                                         ;       GET NEXT COMMAND LINE FROM USER
                                         ;
                                         ; INPUTS:
                                         ;       NONE
                                         ; OUTPUTS:
                                         ;       COMMAND LINE RETURNED WITH NORMAL 'GCL' CONTEXT ( C-BIT CLEAR )
                                         ;       ERROR REPORTED TO USER ( OTHER THAN EOF ) ( C-BIT SET )
                                         ;-

005022  004267  000000G                  $VGCL:: JSR     R2,$SAVVR               ; SAVE VOLATILE REGISTERS
005026                                   10$:    GCL$                            ; GET NEXT COMMAND
005032  103415                                   BCS     110$                    ; FAILED TO GET COMMAND LINE
005034  016700  000310'                          MOV     $CLBUF,R0               ; COMMAND LINE RESTS HERE
005040  105710                           20$:    TSTB    (R0)                    ; IS THIS EOS ?
005042  001771                                   BEQ     10$                     ; YES, THEN LINE CONTAINED ONLY SPACES
005044  122027  000040                           CMPB    (R0)+,#40               ; IS THIS CHARACTER A SPACE
005050  001773                                   BEQ     20$                     ; LINE WAS ALL SPACES
005052  126027  177777  000073           30$:    CMPB    -1(R0),#';              ; WAS IT A COMMENT LINE ?
005060  001762                                   BEQ     10$                     ; YES
005062  000241                                   CLC                             ; MAKE SURE THE CARRY IS CLEAR
005064                                           RETURN

                                         ;
                                         ; ERROR HANDLING
                                         ;

005066  122777  000000G 000132'          110$:   CMPB    #IE.EOF,@$CLIOS         ; WAS ERROR EOF ?
005074  001405                                   BEQ     115$                    ; YES, EXIT WITHOUT MESSAGE
005076                                           ERRPT$  ERRV3,$EROUT
005106  000745                                   BR      $VGCL

005110  000167  000000G                  115$:   JMP     $PANIC                  ; TAKE PATH OF LEAST RESISTANCE
```

```
SYMBOL CROSS REFERENCE                                           CREF   04.00

SYMBOL   VALUE              REFERENCES

L.DLC       000003  G       #5-78
L.DLM       000006  G       #5-78
L.DLS       000010  G       #5-78
L.FLG       000000  G       #5-78
L.KRBA      000016  G       #5-78
L.LEN    =  000022  G       #5-78
L.MPF       000022  G       #5-78
L.NMST      000020  G       #5-78
L.NSTA      000014  G       #5-78
L.OWNR      000021  G       #5-78
L.UNT       000013  G       #5-78
MACTYP      001034 RG       #7-214
MAXOV       000476 RG       #6-126
MS.ALL   =  000437  G       #15-489
MS.CIR   =  001000  G       #15-494
MS.CSR   =  000001  G       #15-484
MS.EXT   =  000200  G       #15-492
MS.INC   =  000100  G       #15-491
MS.LLC   =  000040  G       #15-490
MS.MTP   =  000010  G       #15-487
MS.MUX   =  000420  G       #15-488
MS.PRI   =  000002  G       #15-485
MS.SEC   =  000400  G       #15-493
MS.VCT   =  000004  G       #15-486
M$$PRO   =  ******           4-1           17-781       17-798       17-807
NAMES       001426 R         12-377       #13-390
NONMAP      001000 RG       #7-189
NTPNAM      000734 RG       #7-174
NTPPAR      000740 RG       #7-175
NTPSAV      000756 RG       #7-180
NTPSIZ      000750 RG       #7-178
N.DVNM   =  000032           6-102         6-114
N.FNAM   =  000006           6-102         6-114
N.FTYP   =  000014           6-102         6-114
N.UNIT   =  000034           6-102         6-114
OF$INS   =  000001          #5-80
OF$LOG   =  000100          #5-80
OF$OFF   =  000002          #5-80
OF$ON    =  000000          #5-80
OF$SMC   =  000200          #5-80
OP$ADD   =  000200          #5-80
OP$ALL   =  000200          #5-80
OP$CCS   =  000004          #5-80
OP$CIE   =  004000          #5-80
OP$CON   =  000400          #5-80
OP$COP   =  000001          #5-80
OP$COS   =  000100          #5-80
OP$DST   =  000200          #5-80
OP$DTE   =  000040          #5-80
OP$DUP   =  000002          #5-80
OP$EST   =  000100          #5-80
OP$EVE   =  010000          #5-80
```

Table of contents

```
     5-   55     MACRO DEFINITIONS
     6-   68     FUNCTION TABLES
     7-  163     LOCAL DATA
     8-  196     ERROR MESSAGES
     9-  249     $NLRQB - PROCESS REQUEST BLOCK
    10-  296     NOP - NO PRELIMINARY WORK REQUIRED
    11-  304     RNTI,SCEX - CHECK SET SYS
    12-  342     CEXLD - CHECK COMM EXEC LOADED
    13-  362     PSILD - CHECK PSI LOADED
    14-  381     CHKNOD - CHECK NODE PARMS
    15-  453     PDVA - FIND PDV ADDRESS
    16-  485     CHKPP - CHECK PROCESS PARMS
    17-  523     SPRO - CHECK SET PROCESS PARMS
    18-  550     LLIN - CHECK FOR LINE IN SYSTEM FOR LOGGING
    19-  573     SLTA - GET SLT ADDRESS
    20-  645     CHKLP - CHECK LINE PARAMETERS
    21-  716     CLRPP - CLEAR PROCESS PARMS
    22-  732     SLIN - CHECK SET LINE PARMS
    23-  761     SCIR - CHECK SET CIR PARMS
    24-  779     CPRO - CHECK CLEAR PROCESS PARMS
    25-  909     CLIN - CHECK CLEAR LINE PARMS
    26-  936     CCEX - CHECK CLEAR SYS PARMS
    27- 1010     $BITS - COPY BITS TO $FLAGS
    28- 1034     $FPDV - FIND PDV
    29- 1066     $FSLT - FIND SLT
    30- 1125     $RERR - PRINT ERROR MESSAGE
    31- 1163     REP7 - REPLACE TASK NAME
    32- 1178     REP7A - REPLACE TASK NAME AND CHANNEL NUMBER
    33- 1210     DEV7 - REPLACE TASK AND DEVICE NAMES
    34- 1236     REP7B - REPLACE TASK NAME AND LINE ID
    35- 1289     DEQNTI - DEALLOCATE BLOCK QUEUED TO NTINIT
    438 000416  120360  000001          20$:    CMPB    R3,1(R0)                        ; IS THIS LINE ASSIGNED TO XPT ?
    439 000422  001016                          BNE     131$                            ; IF NE, NO .. ERROR
    440 000424  111000                          MOVB    (R0),R0                         ; ELSE, GET CHANNEL NUMBER
    441 000426  010067  000044G                 MOV     R0,RQB+LO.NAM                   ; STORE CHANNEL NUMBER IN SAFE PLACE
    442 000432  000241                  30$:    CLC                                     ; INDICATE SUCCESS !
    443 000434                          40$:    RETURN
    444                                 ;
    445                                 ; ERROR CONDITIONS
    446                                 ;
    447 000436                          101$:   EMSG$R  01                              ; XPT NOT FOUND
    448 000444                          111$:   EMSG$R  02                              ; XPT NOT LOADED
    449 000452                          121$:   EMSG$R  03                              ; REVERSE MAPPING TABLE IS MESSED UP
    450 000460                          131$:   EMSG$R  04                              ; LINE NOT ASSIGNED TO XPT
    451 000466                          141$:   EMSG$R  05                              ; LINE PARAMETER NODE APPLICABLE
```

```
002050  022721  100000             20$:    CMP     #100000,(R1)+       ; IS CHANNEL 1-N ASSIGNED?
002054  001431                             BEQ     30$                 ; IF EQ, NO
002056  016746  000000G                    MOV     $SLTMA,-(SP)        ; SAVE START OF SLT MAPPING TABLE
002062  012746  002124'                    MOV     #25$,-(SP)          ; TO ALLOW CO-ROUTINE CLEANUP
002066  004267  000000G                    JSR     R2,$SAVVR           ; SAVE R0,R1,R2 (CO-ROUTINE)
002072  014101                             MOV     -(R1),R1            ; GET SYSTEM LINE NUMBER (BYTE)
002074  042701  177400                     BIC     #^C<377>,R1         ;       ...
002100  006301                             ASL     R1                  ; CONVERT TO A WORD INDEX
002102  060166  000012                     ADD     R1,10.(SP)          ; ADJUST SLT POINTER
002106  017666  000012   000012            MOV     @10.(SP),10.(SP)    ; POINT AT SLT
002114  062766  000000G  000012            ADD     #L.FLG,10.(SP)      ; POINT AT FLAGS WORD
002122                                     RETURN                      ; RESTORE R0,R1,R2; RETURN TO 25$
002124  032736  000000G            25$:    BIT     #LF.ACT,@(SP)+      ; IS CHANNEL 1-N ACTIVE?
002130  001403                             BEQ     30$                 ; IF EQ, NO
002132                                     EMSG$   Z3                  ; CHANNEL STILL ACTIVE (SETS R0 NON-ZERO)
002140  020102                     30$:    CMP     R1,R2               ; SCAN ENTIRE CHANNEL TABLE
002142  001342                             BNE     20$                 ;       ...
002144                             40$:
002144  005400                             NEG     R0                  ; ANY ERRORS?
002146  103550                             BCS     150$                ; IF CS, YES
002150  016701  000000G                    MOV     $PDVA,R1            ; GET PDV ADDRESS
002154  016767  000000G  000000G           MOV     $DEVHD,$BFR         ; AND RSX11M DEVICE POINTER
002162  012767  000036   000000G           MOV     #D.PCB+2,$RSIZE     ; SET UP TRANSFER LENGTH
002170  012702  000000G                    MOV     #$BFR,R2            ; POINT AT VIRTUAL IMAGE
002174  011267  000000G            50$:    MOV     (R2),$RADDR         ; SET NEXT TRANSFER ADDRESS
002200  000241                             CLC                         ; CLEAR C-BIT
002202  001532                             BEQ     150$                ; IF EQ, ALL DONE
002204                                     GETRV                       ; READ IN DCB IMAGE
002210  026162  000000G  000034            CMP     Z.PCB(R1),D.PCB(R2) ; IS OUR LLC ALSO THIS DEVICE?
002216  001366                             BNE     50$                 ; IF NE, NO
002220  016203  000002                     MOV     D.UCB(R2),R3        ; GET FIRST UCB ADDRESS
002224  116267  000006   000244'           MOVB    D.UNIT(R2),UNIT     ; AND FIRST UNIT NUMBER
002232  016746  000000G            60$:    MOV     $RADDR,-(SP)        ; SAVE CURRENT DCB ADDRESS
002236                                     SWTCHI  #UCBBUF             ; POINT AT UCB IMAGE BUFFER
002244                                     GETRV   R3,D.UCBL(R2)       ; READ IN UCB IMAGE
002262  012703  000252'                    MOV     #UCBBUF,R3          ; POINT AT UCB IMAGE
002266  012667  000000G                    MOV     (SP)+,$RADDR        ; RESTORE DCB ADDRESS
002272  032763  100000   000010            BIT     #DV.MNT,U.CW1(R3)   ; IS THIS A MOUNTABLE DEVICE ?
002300  001407                             BEQ     70$                 ; IF EQ, NO - DON'T CHECK FOR MOUNTED
002302  132763  000100   000005            BITB    #US.MNT,U.STS(R3)   ; IS UNIT MOUNTED?
002310  001003                             BNE     70$                 ; IF NE, NO
002312                                     EMSG$R  Z4                  ; DEVICE STILL MOUNTED
002320  066203  000010             70$:    ADD     D.UCBL(R2),R3       ; GET NEXT UCB ADDRESS
002324  105267  000244'                    INCB    UNIT                ; AND NEXT UNIT NUMBER
002330  126267  000007   000244'           CMPB    D.UNIT+1(R2),UNIT   ; ALL UNITS CHECKED?
002336  103335                             BHIS    60$                 ; IF HIS, NO
002340  005062  000012                     CLR     D.DSP(R2)           ; MARK DEVICE ''UNLOADED''
002344  005062  000034                     CLR     D.PCB(R2)           ;       ...
002350                                     SWTCHO                      ; RESET DEFAULT BUFFER
002356                                     PUTRC   ,#D.PCB+2           ; REWRITE DCB TO SYS IMG
002370  000437                             BR      150$                ; AND GET OUT

002372  012701  000000G            100$:   MOV     #L.DLC,R1           ; ASSUME DLC
002376  032767  000000G  000000G           BIT     #FL.DLC,$FLAGS      ; IS IT A DLC?
002404  001002                             BNE     110$                ; IF NE, YES
002406  062701  000000G                    ADD     #L.DDM-L.DLC,R1     ; IT'S A DDM
002412  016702  000000G            110$:   MOV     $SLTMA,R2           ; GET SLT MAPPING TABLE ADDRESS
```

```
                                                                                         D  1
VNTI - VNP NETWORK INITIALIZATI MACRO V05.03b Saturday 29-Jun-85 02:46  Page 110-6
Symbol table

$SDBNM= ****** GX       $$$ARG= 000003          .HSTAD  007416R         .PLGTH= ****** GX       .RNUM   011330R
$SDBS   002776RG        $$$FLG= 177777          .HSTAR  007470R         .PNUMB= ****** GX       .RRTM   012110R
$SDBSZ= ****** GX       $$$KEY= 000000          .MGEXT= ****** GX       .PNUMH= ****** GX       .R1     003632R
$SLTA = ****** GX       $$$OST= 000014          .NAM    004464R         .PPRO = ****** GX       .R2     003642R
$SLTMA= ****** GX       $$$STA= 000000          .NAM1   004452R         .PROMF  006520R         .R3     003652R
$SLTNM= ****** GX       $$$TMP= 000411R     005 .NAM2   004460R         .PRONM  006500R         .SBPCB= ****** GX
$STRNG= 000004          .ALLC   004562R         .NODAR  007312R         .PSTCN= ****** GX       .SBRA   007030R
$SUBXP= 000010          .ALLOC= ****** GX       .NODNB  007236R         .PSTPT= ****** GX       .SCTIM  007002R
$SWOVL= ****** GX       .BASEB= 140000          .NODNM  007056R         .RADDR  011674R         .SCTL   006604R
$UMRAL  002262R         .BUFCT  006424R         .NOMAP  004426R         .RAMXC  012412R         .SDBNM= ****** GX
$VFLAG= ****** GX       .BUFDF  006324R         .N1     003662R         .RAMXH  012340R         .SDBSZ= ****** GX
$VNTI   000000RG        .BUFNM  006356R         .N2     003672R         .RARE   011614R         .SLTDF  006562R
$VRQB = ****** GX       .BUFSK  006342R         .N3     003710R         .RBEA   012250R         .STADF  006576R
$XAVL = ****** GX       .CRCOV= ****** GX       .N4     003726R         .RBTM   012464R         .STRNG  007706R
$XBIAS= ****** GX       .FLBK0  016372RG        .OCPY   013434RG        .RCDST  011746R         .SUNT   006614R
$XLINK= ****** GX       .FLBK1  016410RG        .OEND   013456RG        .RDBNM= ****** GX       .SYSID  007574R
$XPAR = ****** GX       .FLBK2  016426RG        .OFLAG  013402RG        .RDBSZ= ****** GX       .SYSNM= ****** GX
$$LINX= ****** GX       .FLBK3  016444RG        .ONAME  013306RG        .RHOP   012022R         .TPARS= ****** GX
$$NODC= ****** GX       .FLBK4  016462RG        .ONUM   013242RG        .RMAXA  012160R         .$$$$.= 000034
$$SHFT= 000001          .FUDG   004056R         .OVFY   013356RG        .RNAME  011254R

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        020166    001   (RW,I,LCL,REL,CDN)
DATA    002552    002   (RW,D,LCL,REL,CDN)
$STATE  002552    003   (RW,D,LCL,REL,CON)
$KTAB   000120    004   (RW,D,LCL,REL,CON)
$KSTR   000420    005   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 259
Work  file writes: 237
Size of work file: 39040 Words  ( 153 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:08:00.08
SY:VNTI.V2,[132,134]VNTI/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VNTI
```

```
                                                                      E  1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50        PAGE 1
```

```
                                                                      D  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 13
SYMBOL  CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

RQB      = ******  GX       *8-366    *8-367     8-369    *16-963    16-965   *107-4777 *107-4784 *107-4785 *107-4786
                            *107-4787 *107-4788  107-4790 *108-4815 *108-4822 *108-4823 *108-4824 *108-4825 *108-4826
                             108-4828 *109-4849 *109-4856 *109-4857 *109-4858 *109-4859 *109-4860  109-4862
R$$MPL   = ******            5-75      5-75      5-75      8-251     8-261     8-343     8-426     11-720    12-780
                             12-789    16-980    17-1056   26-1562
R$$11D   = ******            5-72      5-75
R$$11M   = 000000            5-72
R$$11S   = ******            5-72
R.ADD      000010           #5-64     *69-3170  *69-3171   69-3176   88-3763
R.FLAG     000012           #5-64     *69-3172
R.LEN      000014           #5-64      69-3162   69-3175   88-3762
R.LNK      000000           #5-64
R.NAM      000002           #5-64     *69-3167  *69-3168  *69-3169
SBFLG    = ******  GX       *8-233     8-477     30-1832
SBNAM    = ******  GX        26-1549   26-1550
SBPAR    = ******  GX        17-1063   17-1065
SBPCB    = ******  GX       *8-234    *8-485     16-974
SBPOOL   = ******  GX       *8-232    *8-482     17-1045   17-1061   26-1545   26-1619   26-1631
SBUF       001776 R         *10-581   *10-582    10-588    10-596   #10-619
SEG        011220 R         #67-3110
SERADD     000140 RG        #7-163     93-3990   94-4045   94-4046   94-4047
SERALL     014640 R          93-4010  #94-4025
SERBEG     000110 RG        #7-152     92-3966
SERCIR     000116 RG        #7-156
SERCR      015356 RG        #96-4159
SERCUR     000256 RG        #7-183    *94-4031   94-4067   103-4528
SERDCT     000136 RG        #7-162    *91-3917   94-4058
SERDEV     000120 RG        #7-157    *91-3908   94-4050
SERDMP     000132 RG        #7-161    *91-3912  *91-3913   94-4056   94-4057
SERDV      014414 RG        #91-3906
SEREND     014620 RG        #93-4010
SERFIN     000266 RG        #7-188     92-3968
SERFLG     000110 RG        #7-153     94-4041  *103-4494 *103-4498 *103-4502 *103-4506 *103-4510  103-4531
SERHST     000114 RG        #7-155    *91-3900  *91-3903   94-4043
SERINI     014532 RG        #92-3966
SERMUL     014314 RG        #91-3882
SERMUX     014324 RG        #91-3885
SERNOD     000112 RG        #7-154    *91-3891  *91-3894   94-4042
SEROFF     000260 RG        #7-184    *103-4493 *103-4497 *103-4501 *103-4505 *103-4509  103-4533
SERPAS     000122 RG        #7-159     94-4051   94-4052   94-4054   94-4055   95-4112
SERSLN     014460 RG        #92-3935
SERSLT     000262 RG        #7-186    *92-3948   94-4048  *97-4232
SERTMP     000146 RG        #7-164     93-4002
SERUNI     014304 RG        #91-3879
SETBLK     016040 R         #101-4387  103-4512
SETST      016674 RG        #103-4539
SF.ACT   = 000200           #5-73
SF.CIR   = 000010           #5-81
SF.DEV   = 000020           #5-81
SF.DIA   = 004000           #5-81      103-4506
SF.DPA   = 000100           #5-81
SF.DPC   = 000200           #5-81
```

```
                                                                      E  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 14
```

D 3

```
**FILE**ID**VPIO
```

```
VV      VV   PPPPPPPP        IIIIII       000000
VV      VV   PPPPPPPP        IIIIII       000000
VV      VV   PP      PP        II       00      00
VV      VV   PP      PP        II       00      00
VV      VV   PP      PP        II       00      00
VV      VV   PP      PP        II       00      00
VV      VV   PPPPPPPP          II       00      00
VV      VV   PPPPPPPP          II       00      00
VV      VV   PP                II       00      00
VV      VV   PP                II       00      00
  VV  VV     PP                II       00      00       ....
  VV  VV     PP                II       00      00       ....
    VV       PP              IIIIII       000000         ....
    VV       PP              IIIIII       000000         ....
```

```
LL              SSSSSSSS   TTTTTTTTTT
LL              SSSSSSSS   TTTTTTTTTT
LL            SS               TT
LL            SS               TT
LL            SS               TT
LL            SS               TT
LL              SSSSSS         TT
LL              SSSSSS         TT
LL                    SS       TT
LL                    SS       TT
LL                    SS       TT
LL                    SS       TT
LLLLLLLLLL    SSSSSSSS         TT
LLLLLLLLLL    SSSSSSSS         TT
```

E 3

VPIO - VNP PROCESS I/O ROUTINES MACRO V05.03b Wednesday 28-Aug-85 20:03

```
                                                                       D  4
VPIO - VNP PROCESS I/O ROUTINES MACRO V05.03b Wednesday 28-Aug-85 20:03  Page 11-1
CR.MC - CREATE MICROCODE PARTITION

    461 000770  010167  000000G                  MOV     R1,$LLEN        ; STORE SIZE IN BYTES
    462 000774  010167  000144'                  MOV     R1,BYTES        ; ...
    463
    464 001000                          10$:     RETURN
    465
    466
```

```
                                                                       E  4
VPIO - VNP PROCESS I/O ROUTINES MACRO V05.03b Wednesday 28-Aug-85 20:03  Page 12
```

```
                                                                                   D 5
VPIO - VNP PROCESS I/O ROUTINES MACRO V05.03b Wednesday 28-Aug-85 20:03  Page 17-2
Symbol table

P.BLKS  000016         SYSFDB= ****** GX      T$NLEN  000010         UNAMC   000154R   002  $PARHD= ****** GX
P.BUSY  000024         S$$WRG= 000000         T$NNUL  000002         UNAOP   000050R   002  $PDVNM= ****** GX
P.IOC   000003         S$$YSZ= 007600         T$NOPL  000006         UNARE   000136R   002  $PDVTA= ****** GX
P.LNK   000000         TMPPCB  000152R   002  T$NRNI  000042         VF.SS = ****** GX      $PNAME  000254RG  002
P.MAIN  000012         TPS   = 177564         T$NRPL  000005         V$$CTR= 001000         $PRCLN= ****** GX
P.NAM   000004         TS$NHD= ****** GX      T$NRUL  000007         X$$DBT= 000000         $PREXT= ****** GX
P.OWN   000026         T$FLAG  000044         T$NVR   000001         Z.NAM = ****** GX      $PRINC= ****** GX
P.PRI   000002         T$LIF   000013         T$RPRI  000040         $ALL15= ****** GX      $PRIO   000000RG
P.REL   000014         T$LIFL  000013         T$SVC   000034         $BFR  = ****** GX      $PRLLN= ****** GX
P.SIZE  000016         T$LIFO  000013         T$T5    000030         $CLDEQ= ****** GX      $PUTRC= ****** GX
P.STAT  000030         T$LIFS  000013         T$T6    000032         $CLIOS= ****** GX      $RADDR= ****** GX
P.SUB   000010         T$LIN   000000         T$$KMG= 000000         $CLOPE= ****** GX      $READX= ****** GX
P.SWSZ  000022         T$LIPS  000006         T$$MIN= 000000         $DDMPV= ****** GX      $RLBL = ****** GX
P.TCB   000026         T$LLD   000012         UBMPR = 170200         $DEA16= ****** GX      $SAVAL= ****** GX
P.WAIT  000020         T$LLDC  000045         UDSAR0= 177660         $DECPT= ****** GX      $TIOUT= ****** GX
Q$$OPT= 000010         T$LLDL  000012         UDSDR0= 177620         $DFUIC= ****** GX      $VFLAG= ****** GX
RMCLBL  000726R        T$LLDO  000012         UISAR0= 177640         $ERMSG= ****** GX      $XPAR = ****** GX
R$$DER= 000000         T$LLDS  000012         UISAR4= 177650         $FLAGS= ****** GX      $$    = 177777
R$$K11= 000001         T$LLEN  000046         UISAR5= 177652         $FMASK= ****** GX      $$EXT = ****** GX
R$$SND= 000000         T$LOPR  000002         UISAR6= 177654         $GETRV= ****** GX      $$INC = ****** GX
R$$11M= 000000         T$LTCL  000024         UISAR7= 177656         $ICBLN= ****** GX      $$PAR = ****** GX
SFIL    000624R        T$LTIM  000026         UISDR0= 177600         $IOSB = ****** GX      $$$   = 000054R   002
SISDR0= 172200         T$LTPR  000014         UISDR4= 177610         $LDSIZ  000260RG  002  $$$$  = 000136R   002
SR0   = 177572         T$LTPS  000020         UISDR5= 177612         $LLEN = ****** GX      .PLGTH= ****** GX
SR3   = 172516         T$NAPL  000004         UISDR6= 177614         $MUL  = ****** GX      .PPRO = ****** GX
SWR   = 177570         T$NFE   000000         UISDR7= 177616         $NAME = ****** GX

. ABS.  000104    000   (RW,I,GBL,ABS,OVR)
        003174    001   (RW,I,LCL,REL,CON)
DATA    000402    002   (RW.D,LCL,REL,CON)
..BUF   000002    003   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 15125 Words  ( 60 Pages)
Size of core pool: 16552 Words  ( 63 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:38.04
DB2:VPIO.T34,[132,134]VPIO/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VPIO
```

```
                                                        E 5
VPIO       CREATED BY  MACRO  ON 28-AUG-85 AT 20:04     PAGE 1
```

```
                                              D 6
PRMPTS - VNP PROMPT STRINGS     MACRO V05.03b Saturday 29-Jun-85 02:55  Page 6-2
Symbol table

A$$CHK= 000000        M$$CRX= 000000        P$DTE    000560 G      P$PID    001654 G      R$$SND= 000000
A$$CPS= 000000        M$$FCS= 000000        P$DUP    000600 G      P$PRI    001636 G      R$$11M= 000000
A$$PRI= 000000        M$$MGE= 000000        P$EVE    000614 G      P$SCO    001676 G      S$$WRG= 000000
A$$TRP= 000000        M$$NET= 000000        P$GRP    000646 G      P$SEAL   001732 G      S$$YSZ= 007600
C$$CKP= 000000        M$$OVR= 000000        P$HOS    000666 G      P$SECI   001760 G      T$$KMG= 000000
C$$ORE= 000400        N$$ACC= 000001        P$HTM    000724 G      P$SELI   002010 G      T$$MIN= 000000
C$$RSH= 177564        N$$BUF= 000001        P$LGID   001102 G      P$SELO   002036 G      V$$CTR= 001000
D$$BUG= 177514        N$$LDV= 000001        P$LIST   001032 G      P$SEMO   002114 G      X         000000RG     002
D$$ISK= 000000        N$$MCP= 000001        P$LNID   000752 G      P$SENO   002066 G      X$$DBT= 000000
D$$L11= 000001        N$$MLL= 000001        P$LOC    001052 G      P$SEOB   002144 G      Y         000000RG     003
D$$YNC= 000000        N$$MOV= 000010        P$LUST   001130 G      P$SEPR   002174 G      $ALPHA= 000022
D$$YNM= 000000        N$$NCT= 000001        P$LTM    001002 G      P$SET    001710 G      $ANY   = 000020
E$$XPR= 000000        N$$PEM= 000001        P$MAX    001152 G      P$SHO    002224 G      $BLANK= 000006
F$$LVL= 000001        P$ACC    000010 G     P$MOID   001250 G      P$SIN    002250 G      $DIGIT= 000024
G$$TPP= 000000        P$ACT    000024 G     P$MUL    001302 G      P$SNAM   002306 G      $DNUMB= 000014
G$$TSS= 000000        P$ADD    000056 G     P$MXCO   001176 G      P$STA    002324 G      $EOS   = 000012
G$$TTK= 000000        P$AID    000072 G     P$MXLI   001226 G      P$TER    002336 G      $EXIT  = 000000
G$$WRD= 000000        P$CLCI   000146 G     P$NAD    001334 G      P$TPAS   002372 G      $FAIL  = 177777
I$$RAR= 000000        P$CLE    000200 G     P$NAM    001422 G      P$TRIB   002414 G      $GPRM  = 000001
I$$RDN= 000000        P$CLLI   000224 G     P$NET    001356 G      P$USE    002442 G      $LAMDA= 000000
K$$CNT= 177546        P$CLLO   000254 G     P$NID    001372 G      P$VEC    002472 G      $NUMBR= 000002
K$$CSR= 177546        P$CLNO   000306 G     P$NOST   001440 G      P$VEST   002506 G      $RAD50= 000016
K$$LDC= 000000        P$CLOB   000336 G     P$OBNA   001460 G      P$VNP    000000 G      $RONLY= 000001
K$$TPS= 000074        P$COP    000426 G     P$OBUS   001522 G      P$X5ST   002536 G      $STRNG= 000004
LD$LP = 000000        P$COS    000370 G     P$OBVE   001542 G      P$$P45= 000000        $SUBXP= 000010
L$$ASG= 000000        P$CRID   000112 G     P$OID    001500 G      P$$WRD= 000000        $$PRMP    000000RG     005
L$$DRV= 000000        P$CSR    000442 G     P$OWN    001572 G      Q$$OPT= 000010        $$$FLG= 177777
L$$P11= 000001        P$DEA    000462 G     P$PAR    001612 G      R$$DER= 000000        $$$KEY= 177777
L$$11R= 000000        P$DES    000512 G     P$PEN    000402 G      R$$K11= 000001        $$$LEN= 000024
M$$CRB= 000124        P$DST    000540 G

. ABS.   002564    000   (RW,I,GBL,ABS,OVR)
         000000    001   (RW,I,LCL,REL,CON)
$STATE   000000    002   (RO,D,LCL,REL,CON)
$KTAB    000000    003   (RO,D,LCL,REL,CON)
$KSTP    000000    004   (RO,D,LCL,REL,CON)
$PRMPT   002564    005   (RO,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file  9783 Words  ( 39 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:14.50
SY:VPRMPT.V2,[132,134]VPRMPT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VPRMPT
```

```
                                               E  6
VPRMPT      CREATED BY  MACRO  ON 29-JUN-85 AT 02:55      PAGE 1
```

```
                                        .SBTTL  AXSYM - AUX SPECIAL SYMBOLS
                                ;+
                                ; *** - AXSYM - AUX SPECIAL SYMBOLS
                                ;
                                ; INPUTS:
                                ;       $NAME = RAD50 PROCESS NAME
                                ;       $BIAS = PROCESS BIAS
                                ;       $SQSRV = ADDRESS OF $SQSRV ROUTINE IN AUX PROCESS
                                ;       PR100M - ADDRESS OF 100 MSEC TIMER SERVICE ROUTINE IN AUX PROCESS
                                ;       PR1SC = ADDRESS OF 1 SEC TIMER SERVICE ROUTINE IN AUX PROCESS
                                ;
                                ; OUTPUTS:
                                ;       C-BIT = SUCCESS/FAILURE
                                ;       FORK BLOCK AND TIMER BLOCKS INITIALIZED
                                ;
                                ;-

000460                          AXSYM:
000460  022767  004640  000000G         CMP     #^RAUX,$NAME            ; IS THIS PROCESS AUX ?
000466  000241                          CLC                             ; ASSUME NOT
000470  001042                          BNE     20$                     ; IF NE, NO .. DO NOTHING

000472  005767  000000G                 TST     .MGMGE                  ; MAPPED SYSTEM ?
000476  100411                          BMI     5$                      ; IF MI, NO
                                ;
                                ; MAPPED SYSTEMS
                                ;
000500  016767  000000G 000010G         MOV     $BIAS,$CMFRK+10         ; STORE BIAS FOR AUX PROCESS
000506  016767  000000G 000014G         MOV     $BIAS,$T100C+14         ; ...
000514  016767  000000G 000014G         MOV     $BIAS,$T1SCL+14         ; ...
                                ;
                                ; COMMON PATH
                                ;
000522  005767  000130'         5$:     TST     SQBLK+2                 ; WAS $SQSRV DEFINED ?
000526  001422                          BEQ     10$                     ; IF EQ, NO .. ERROR
000530  016767  000136' 000000G         MOV     $SQSRV,$CMFRK           ; ELSE, INITIALIZE FORK BLOCK
000536  016767  000136' 000002G         MOV     $SQSRV,$CMFRK+2         ; ...
000544  005767  000142'                 TST     MILBLK+2                ; WAS PR100M DEFINED ?
000550  001411                          BEQ     10$                     ; IF EQ, NO .. ERROR
000552  016767  000150' 000012G         MOV     PR100M,$T100C+12        ; ELSE, INITIALIZE 100 MSEC CLOCK BLOCK
000560  016767  000162' 000012G         MOV     PR1SC,$T1SCL+12         ; INITIALIZE 1 SEC CLOCK BLOCK
000566  005767  000154'                 TST     SECBLK+2                ; WAS PR1SC DEFINED ?
000572  001001                          BNE     20$                     ; IF NE, YES .. OKAY
000574  000261                  10$:    SEC                             ; ELSE, INDICATE FAILURE
000576                          20$:    RETURN
```

```
    970                                         .SBTTL  $ALMHC - ALLOCATE MINHOPS/MINCOST MATRICES
    971                                 ;+
    972                                 ;
    973                                 ; $ALMHC - ALLOCATE MIN HOPS/MIN COST MATRIX FROM POOL..
    974                                 ;
    975                                 ; INPUTS:
    976                                 ;       $PDVA - XPT'S PDV ADDRESS
    977                                 ;
    978                                 ; OUTPUTS:
    979                                 ;       ALLOCATE MIN HOPS/COST MATRIX
    980                                 ;       N$MHC1   (XPT'S DDB) - NUMBER OF ENTRIES
    981                                 ;       N$MHC1+2 (XPT'S DDB) - BIAS
    982                                 ;       N$MHC1+4 (XPT'S DDB) - APR6 VIRTUAL ADDRESS OF MATRIX
    983                                 ;       N$MHC2   (XPT'S DDB) - NUMBER OF ENTRIES IN AREA MATRIX
    984                                 ;       N$MHC2+2 (XPT'S DDB) - BIAS OF AREA MATRIX
    985                                 ;       N$MHC2+4 (XPT'S DDB) - APR6 VIRTUAL ADDRESS OF MATRIX
    986                                 ;-
    987 003424                          $ALMHC::CALL    $SAVAL                  ; SAVE ALL REGISTERS
    988
    989 003430                                  GETRV   $DECPT,#D$END           ; READ DECNET HOME BLOCK
    990 003450  016702  000060G                 MOV     $BFR+D$NN,R2            ; STORE MAX NODES
    991 003454  001574                          BEQ     90$                     ; IF END NODE BRANCH
    992
    993 003456  010204                          MOV     R2,R4                   ; COPY MAX NODE
    994 003460  005204                          INC     R4                      ; INCLUDE EXTRA NODE ENTRY
    995 003462  016703  000062G                 MOV     $BFR+D$NA,R3            ; GET MAX AREAS
    996 003466  001402                          BEQ     10$                     ; IF NONE - BRANCH
    997 003470  005204                          INC     R4                      ; INCLUDE EXTRA AREA
    998 003472  060304                          ADD     R3,R4                   ; CALCULATE TOTAL ENTRIES
    999 003474  006304                  10$:    ASL     R4                      ; CONVERT TO BYTE COUNT (1 WORD/NODE)
   1000 003476  010401                          MOV     R4,R1                   ; COPY BYTE COUNT
   1001 003500  062701  000077                  ADD     #77,R1                  ; ROUND TO BLOCK BYTE BOUNDARY
   1002 003504  042701  000077                  BIC     #77,R1                  ; ...
   1003 003510                                  ASR$    6,R1                    ; CONVERT TO BLOCK COUNT
   1004 003524  010205                          MOV     R2,R5                   ; SAVE MAX NODES
   1005 003526                                  CALL    $ALN18                  ; ALLOCATE MATRIX FROM NON-UMR SPACE
   1006 003532  103546                          BCS     100$                    ; BR IF ERROR
   1007 003534  010502                          MOV     R5,R2                   ; RESTORE MAX NODES
   1008 003536  016705  000000G                 MOV     $PDVA,R5                ; GET XPT'S PDV ADDRESS
   1009 003542  016505  000000G                 MOV     Z.DAT(R5),R5            ; GET XPT'S DDB ADDRESS
   1010 003546                                  GETRV   R5,#N$XLEN              ; READ IN XPT'S DDB
   1011 003564  010201                          MOV     R2,R1                   ; GET NUMBER OF ENTRIES IN NODE HOPS/COST
   1012 003566  005201                          INC     R1                      ; INCREASE BY ONE
   1013 003570  010167  000036G                 MOV     R1,N$MHC1+$BFR          ; STORE NUMBER OF ENTRIES IN MATRIX
   1014 003574  006301                          ASL     R1                      ; CONVERT TO BYTES
   1015
   1016 003576  010067  000040G                 MOV     R0,N$MHC1+2+$BFR        ; STORE APR BIAS FOR MIN HOPS/MIN COST
   1017 003602  005004                          CLR     R4                      ; POINT TO START OF MATRIX
   1018 003604                                  BIAS    R4                      ; ... (XPT NEEDS APR6 BIAS)
   1019 003614  010467  000042G                 MOV     R4,N$MHC1+4+$BFR        ; STORE START OF MATRIX
   1020 003620  060104                          ADD     R1,R4                   ; POINT TO START OF AREA HOPS/COST
   1021 003622  010301                          MOV     R3,R1                   ; GET NUMBER OF ENTRIES IN AREA HOPS/COST
   1022 003624  001410                          BEQ     20$                     ; IF NONE - BRANCH
   1023
   1024 003626  005201                          INC     R1                      ; INCREASE BY ONE
   1025 003630  010167  000044G                 MOV     R1,N$MHC2+$BFR          ; STORE NUMBER OF ENTRIES IN AREA MATRIX
   1026 003634  010067  000046G                 MOV     R0,N$MHC2+2+$BFR        ; STORE APR BIAS FOR AREA MATRIX
```

```
                                                     D 9
VPRO      CREATED BY  MACRO  ON 28-AUG-85 AT 20:05   PAGE 5
SYMBOL CROSS REFERENCE                               CREF  04.00

SYMBOL  VALUE             REFERENCES

T$SVC     000034          #5-68
T$T5      000030          #5-68    *11-586
T$T6      000032          #5-68    *11-587
VECLEN    000006 R        #6-94
VECSYM    000044 R        #6-142
VECTAB    000000 R        #6-91
VECVAL    000054 R        #6-144    8-344     10-489
VLNSYM    000056 R         6-142   #6-148
VLNVAL    000066 R        #6-150    8-345
VLSYM     000062 R        #6-149
VTSYM     000050 R        #6-143
V.ADD     000006          *8-344
V. LG     000002          *8-343
V. EN     000012           8-338    8-351
V.LNK     000000          *8-340
V.PDV     000004          *8-348
V.SIZ     000010          *8-345
XBLK1     000164 R         6-185   #6-192     10-486
XBLK2     000176 R         6-192   #6-195
XBLK3     000210 R         6-195   #6-198
XPTDSP    000016 R        #6-113   *11-569   *13-718    13-723    13-739   *15-874    15-880
XPTPDV    000014 R        #6-112   *11-526    11-560    11-567
Z.AVL   = ******  GX      *7-303
Z.DAT   = ******  GX       11-570   12-639    13-719    13-787    14-835    14-845    15-875    15-923    16-1009
                           16-1064  17-1110

Z.DSP   = ******  GX      *7-276   *7-286    *7-288     11-569    13-718    15-874
Z.LLN   = ******  GX       14-815   15-922
Z.MAP   = ******  GX       11-561
Z.NAM   = ******  GX       11-557   18-1169
Z.PCB   = ******  GX       7-249   *7-269
$ALADJ    002440 RG       #14-812
$ALECL    004056 RG       #17-1095
$ALMHC    003424 RG       #16-987
$ALN18  = ******  GX       12-635   14-833    14-842    15-939    16-1005   17-1107
$ALPOL  = ******  GX       8-339
$ALTLC    001062 RG       #11-523
$AZFS   = ******  GX       7-258    10-477
$BFR    = ******  GX       7-272    7-285     7-299    *7-309    *7-310    *8-340    *8-343    *8-344    *8-345
                          *8-348    11-573    11-574    11-576    11-578    11-580   *11-585   *11-586   *11-587
                          *11-588  *11-589    12-619    12-622    12-624   *12-641   *12-645   *12-648   *12-654
                           12-655  *12-655    12-657   *12-658    12-677    13-708    13-712   *13-722    13-724
                           13-745   13-768   *13-789   *13-790    14-818    14-820    14-821    14-822   *14-836
                          *14-839  *14-846    15-878    15-879    15-894    15-911   *15-929   *15-945   *15-948
                           15-952   16-990    16-995   *16-1013  *16-1016  *16-1019  *16-1025  *16-1026  *16-1027
                          *16-1030 *16-1031  *16-1032   16-1038   16-1039   16-1040   16-1047   16-1066   16-1068
                           16-1069  17-1098  *17-1112  *17-1118   17-1125  *17-1142
$BIAS   = ******  GX      *7-277   *7-287     7-308     7-311     9-384     9-385     9-386
$CACH     002000 RG       #13-707
$CAT5   = ******  GX       10-463   10-465    10-472
$CEACC  = ******  GX       8-350
$CETDF  = ******  GX       8-347
$CKT      002720 RG       #15-873
```

```
                                                     E 9
VPRO      CREATED BY  MACRO  ON 28-AUG-85 AT 20:05   PAGE 6
```

```
    204 000000                                  $BFR::  .BLKB   512.            ; DEFAULT VIRTUAL I/O BUFFER
    205 001000                                  LINBUF::.BLKB   128.            ; CONFIG FILE SCANNER BUFFER
    206
    207 000000                                          .PSECT  TMP$$B          ; END OF TEMPLATE PROCESSING BUFFER
    208 000000                                  $TMPND::                        ; END OF BUFFER
    209 001024                                          .PSECT
    210 001024  000000                          LINSIZ::.WORD   0               ; NUMBER OF CHARACTERS IN BUFFER
    211 001026  000000                          CFSFLG::.WORD   0               ; CETAB FILE SCAN FLAG AND DATA BASE
    212 001030  000000                          DFPTR:: .WORD   0
    213 001032  000000                          DFCNT:: .WORD   0
    214 001034  000000                          MACTYP::.WORD   0
    215 001036  000000                          PDVNAM::.WORD   0
    216 001040  000000                          SLTNUM::.WORD   0
```

```
 003570  000000          .SYSNM::.WORD   0               ; RSX SYSTEM NAME ADDRESS
 003572  000000          .CLINS::.WORD   0               ; SYMBOLS
 003574  000000          .FRKHD::.WORD   0               ;  USED FOR
 003576  000000          .HEADR::.WORD   0               ;   CEX
 003600  000000          .STKDP::.WORD   0               ;     VALIDATION
 003602  000000          .PLGTH::.WORD   0               ; LENGTH OF PCB
 003604                  $MXVCT::
 003604  000000          .VCTR::.WORD    0               ; HIGHEST LEGAL INTERRUPT ADDRESS + 4
 003606  000000          .NS0::  .WORD   0               ; ERROR LOG INTERRUPT ( VECTORS 0 - 76 )
 003610  000000          .NS1::  .WORD   0               ; VECTORS 100 - 176
 003612  000000          .NS2::  .WORD   0               ; VECTORS 200 - 276
 003614  000000          .NS3::  .WORD   0               ; VECTORS 300 - 376
 003616  000000          .NS4::  .WORD   0               ; VECTORS 400 - 476
 003620  000000          .NS5::  .WORD   0               ; VECTORS 500 - 576
 003622  000000          .NS6::  .WORD   0               ; VECTORS 600 - 676
 003624  000000          .NS7::  .WORD   0               ; VECTORS 700 - 776
 003626  000000          .MGEXT::.WORD   0               ; 11/70 EXTENDED MEMORY SUPPORT
 003630  000000          .MGMGE::.WORD   0               ; ONLY HERE SO UNMAPPED SYSTEM WILL GET ERROR
 003632  000000          .UMRHD::.WORD   0               ; UMR LISTHEAD
 003634  000000          .DEUMR::.WORD   0               ; ADDRESS OF EXEC URM ROUTINE
 003636  000000          .TSKHD::.WORD   0               ; TASK CONTROL BLOCK LISTHEAD
 003640  000000          .ABTIM::.WORD   0               ; ABSOLUTE TIME COUNTER
 003642  000000          .TLGTH::.WORD   0               ; TCB LENGTH
 003644  000000          .CLKHD::.WORD   0               ; CLOCK QUEUE LISTHEAD
 003646  000000          .PATT:: .WORD   0               ; PCB ATTACH LISTHEAD
 003650  000000          .PHDR:: .WORD   0               ;OFFSET IN PCB TO TASK HEADER ADDRESS
 003652  000000          .PPRO:: .WORD   0               ; PCB OFFSET TO PROTECTION WORD
 003654  000000          .NETPF::.WORD   0               ; NETWORK POWERFAIL RECOVERY ROUTINE ADDRESS
 003656  000000          .NTUIC::.WORD   0               ; NETWORK UIC
                         ; VALUES LOOKED UP IN COMEXEC SYMBOL TABLE


                                 .IF NDF R$$MPL                                          ;[MP01]
 003660  000000          .PDVTA::.WORD   0               ; STARTING ADDRESS OF CETAB
 003662  000000          .CEAVL::.WORD   0               ; CEX POOL LISTHEAD
                                 .ENDC                                                   ;[MP01]

 003664  000000          .DLCOV::.WORD   0               ; DLC PROTOCOL OVERHEAD
 003666  000000          .CRCOV::.WORD   0               ; CRC OVERHEAD
 003670  000000          .CEPWR::.WORD   0               ; POWERFAIL RECOVERY ROUTINE

 003672                  $SYMND::                        ; REFERENCE POINT

                         ; IMAGE OF OFTEN REFERENCED VALUES IN RSX11M.SYS

                         .IF DF  R$$MPL                                                  ;[MP01]
                         $CTLST::.WORD   0               ; CTB LIST HEAD                 ;[MP01]
                         .ENDC                                                           ;[MP01]

 003672  000000          $DEVHD::.WORD   0               ; DEVICE TABLE LISTHEAD
 003674  000000          $EXSIZ::.WORD   0               ; HIGHEST ADDRESS IN EXEC
 003676  000000          $FMASK::.WORD   0               ; SYSTEM FEATURE MASK
 003700  000000          $PARHD::.WORD   0               ; PARTITION CONTROL BLOCK LISTHEAD
 003702  000000          $SYSIZ::.WORD   0               ; SYSTEM SIZE IN 32-WORD BLOCKS
```

```
                                        .SBTTL  ERROR MESSAGES

                                ;
                                ; ERROR MESSAGES
                                ;

005114                                  ERMSG$  <Attempt to reference block number that is not%n>
005173                                  ERMSG$  <within the system image file( %D. ). or EOF detected %n>
005262                                  ERMSG$  <               ( bias:%O/address:%O )>
005314                                  ERBLK$  ERRV1,<$BFR,$RBLCK,$RADDR>
005326                                  ERMSG$  <I/O error accessing system image file ( %E ).%n>
005405                                  ERBLK$  ERRV2,<SYSFDB+F.ERR>
005412                                  ERMSG$  <Failed to get command line.>
005445                                  ERBLK$  ERRV3
005450                                  ERMSG$  <Maximum transfer size with GETRV/PUTRV macros is 512.%n>
005537                                  ERMSG$  <bytes, but specified record size is %D. bytes.>
005615                                  ERBLK$  ERRV4,<$RSIZE>
```

```
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16     PAGE 6
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE             REFERENCES

OP$FIL  = 000200          #5-80
OP$GRP  = 000100          #5-80
OP$HOS  = 000001          #5-80
OP$HTM  = 000002          #5-80
OP$INA  = 004000          #5-80
OP$INC  = 001000          #5-80
OP$KDS  = 002000          #5-80
OP$KDT  = 000400          #5-80
OP$KEV  = 001000          #5-80
OP$KGR  = 001000          #5-80
OP$KNT  = 010000          #5-80
OP$KSK  = 002000          #5-80
OP$LCT  = 000200          #5-80
OP$LNE  = 000040          #5-80
OP$LOC  = 000001          #5-80      #5-80
OP$LST  = 000002          #5-80
OP$LTM  = 000001          #5-80
OP$LVL  = 000010          #5-80
OP$MAC  = 000010          #5-80
OP$MCO  = 000002          #5-80
OP$MDE  = 000400          #5-80
OP$MLI  = 000004          #5-80
OP$MON  = 000100          #5-80
OP$MST  = 000020          #5-80
OP$NAM  = 000004          #5-80      #5-80      #5-80
OP$NET  = 004000          #5-80
OP$NLI  = 000002          #5-80
OP$NOD  = 000020          #5-80
OP$OUT  = 002000          #5-80
OP$OWN  = 000020          #5-80
OP$PAR  = 000010          #5-80
OP$PRI  = 000010          #5-80
OP$RET  = 010000          #5-80
OP$RPA  = 000010          #5-80
OP$SCO  = 000001          #5-80
OP$SEG  = 000400          #5-80
OP$SER  = 000400          #5-80
OP$SIN  = 000010          #5-80
OP$STA  = 000004          #5-80
OP$TPA  = 000020          #5-80
OP$TRI  = 000040          #5-80
OP$UCS  = 000020          #5-80
OP$USE  = 000002          #5-80
OP$VEC  = 000040          #5-80
OP$VER  = 000040          #5-80
OP$XAC  = 000001          #5-80
OP$XPR  = 000002          #5-80
OP$X5S  = 000004          #5-80
OP$X9S  = 000010          #5-80
PBLOX     001004 RG       #7-191
PDVNAM    001036 RG       #7-215
PF.PSV  = 000002  G       #15-624
```

```
        .TITLE  VRQB - VNP PROCESS REQUEST BLOCK
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       VNP - REQUEST BLOCK PARSING ROUTINES
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
;

```

```
                                               .SBTTL  PDVA - FIND PDV ADDRESS
                                       ;+
                                       ; PDVA - FIND PDV ADDRESS
                                       ;
                                       ; INPUTS:
                                       ;       R3=REQUEST BLOCK POINTER
                                       ;               .BLKW   1       ; LR.PRO - PROCESS NAME
                                       ;
                                       ; OUTPUTS:
                                       ;       C-BIT=SUCCESS/FAILURE
                                       ;       $PDVA=PDV ADDRESS
                                       ;       $NAME=RAD50 PDV NAME
                                       ;       $FLAGS SET TO PROCESS TYPE FROM PDV STATUS BYTE
                                       ;-
000474  012300                         PDVA:   MOV     (R3)+,R0        ; GET PROCESS NAME
000476  001417                                 BEQ     101$            ; IF ZERO, RETURN FAILURE
000500  010067  000000G                        MOV     R0,$NAME        ; STORE IT
000504                                         CALL    $FPDV           ; TRY TO FIND PDV
000510  103412                                 BCS     101$            ; IF CS, NOT FOUND
000512  010067  000000G                        MOV     R0,$PDVA        ; SET PDV ADDRESS
000516  116001  000000G                        MOVB    Z.FLG(R0),R1    ; GET PROCESS TYPE FROM PDV STATUS BYTE
000522  042701  000000C                        BIC     #^C<FL.DDM!FL.DLC!FL.LLC>,R1    ;       ...
000526  001406                                 BEQ     111$            ; IF ZERO, IT'S NOT A REAL PDV!
000530  050167  000000G                        BIS     R1,$FLAGS       ; UPDATE NETLDR STATUS
000534                                 10$:    RETURN

                                       ;
                                       ; ERROR CONDITIONS
                                       ;
000536                                 101$:   EMSG$R  97              ; PROCESS NOT IN SYSTEM
000544                                 111$:   EMSG$R  96              ; GARBAGE IN PDV
```

```
    893 002416  016703  000000G                     MOV     $SLTNM,R3         ; AND NUMBER OF SYSTEM LINES
    894 002422  001421                              BEQ     140$              ; IF ZERO, NONE TO CHECK
    895 002424  012246                      120$:   MOV     (R2)+,-(SP)       ; GET FLAGS WORD ADDRESS ON STACK
    896 002426  062716  000000G                     ADD     #L.FLG,(SP)       ; ...
    897 002432  032736  000000G                     BIT     #LF.RDY,@(SP)+    ; IS THIS LINE STILL READY ?
    898 002436  001411                              BEQ     130$              ; IF EQ, NO
    899 002440  016246  177776                      MOV     -2(R2),-(SP)      ; GET SLT ADDRESS ON STACK
    900 002444  060116                              ADD     R1,(SP)           ; ADD INDEX POINTER
    901 002446  123667  000246'                     CMPB    @(SP)+,PDVX       ; DOES THIS LINE REFERENCE OUR PROCESS?
    902 002452  001003                              BNE     130$              ; IF NE, NO
    903 002454                                      EMSG$   Z5                ; LINE STILL READY (SETS R0 NON-ZERO)
    904 002462  005303                      130$:   DEC     R3
    905 002464  001357                              BNE     120$              ; LOOP <SYSTEM LINE COUNT> TIMES
    906 002466  005400                      140$:   NEG     R0                ; SET C-BIT IF ERROR OCCURRED
    907 002470                              150$:   RETURN
```

```
                                                                  E  1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50        PAGE 1
SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL  VALUE               REFERENCES

A$LEN     000004             24-1387
BRADJT    004630 R          #25-1508
BUFUMP  = 172354            #5-75
CB.CCB  = 000002             12-796
CFGKW     000116 RG         #46-2329
CFGST     001754 RG         #46-2329
CFSFLG  = ******   GX       *30-1751     30-1780    *30-1786     30-1788     30-1795    *30-1801     30-1803     30-1807    *30-1814
                             30-1817     30-1824
CF.DDM  = 000002            #5-65
CF.DYN  = 000004            #5-65        54-2604
CF.EIS  = 000010            #5-65
CF.FRK  = 100000            #5-65
CF.LOG  = 000020            #5-65
CF.MDM  = 000001            #5-65
CF.TIM  = 000400            #5-65
CIRMUL    000156 RG         #7-168      *91-3882     92-3948
CIRNAM    000150 RG         #7-166       92-3936     96-4159
CIRNUM    000154 RG         #7-167      *91-3879     92-3943
CIRUNT    000254 RG         #7-182      *91-3885     92-3945
CLASS     000042 R          #6-123      *77-3442     79-3497     79-3510
CL.AST  = 004000            #5-70
CL.ATT  = 000200            #5-70
CL.CAS  = 020000            #5-70
CL.CMT  = 000001            #5-70
CL.EIF  = 000040            #5-70
CL.EXN  = 040000            #5-70
CL.EXT  = 000010            #5-70
CL.IND  = 000002            #5-70
CL.LCC  = 000020            #5-70
CL.NPR  = 000400            #5-70
CL.OPN  = 000100            #5-70        28-1684
CL.RNE  = 000004            #5-70
CL.XIT  = 100000            #5-70
CMODE   = 140000            #5-75
CMPSTR    015740 R           98-4281    #99-4303     99-4310     99-4319
CM$SET  = ******   GX        107-4779    108-4817    109-4851
CRFNBK    016154 R          #102-4431    103-4516
CRSUB     004640 R           17-1103    #26-1534
CTL       000060 R          #6-127      *59-2677     59-2688
CURUNM    000004 R          #6-116      *69-3173     69-3177    *79-3504     79-3511    *80-3537     80-3538    *80-3539     80-3541
                            *86-3684     86-3692    *87-3721     87-3722    *87-3723     87-3725    *88-3753     88-3754    *88-3755
                             88-3761
C.BID     000003            *12-796
DADDR     014432 RG         #91-3912
DCT       014450 RG         #91-3917
DELF      010676 R          #67-3063
DELW      010750 R          #67-3072
DEV       017010 RG         #103-4589
DEVBUF    000242 RG         #7-179       7-180       101-4395    101-4397    103-4543   *103-4591    103-4592   *103-4598
DEVNM     000160 RG         #7-169
DFCNT   = ******   GX       *50-2541    *51-2556    *52-2568
DFPTR   = ******   GX       *50-2540    *51-2555     52-2570    *52-2572
```

```
                                                                  F  1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:5C        PAGE 2
```

```
                                                                   E  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50     PAGE 14

SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL   VALUE              REFERENCES

SF.DUM  = 010000            #5-81       103-4510
SF.ENA  = 000100            #5-73       59-2708    66-3000    66-3004
SF.HAD  = 000004            #5-81
SF.HST  = 000002            #5-81
SF.LOA  = 002000            #5-81       103-4502
SF.LPB  = 000004            #5-73
SF.MFL  = 000040            #5-73       108-4813   108-4832
SF.PAC  = 000020            #5-73
SF.PH3  = 020000            #5-81
SF.PSS  = 000040            #5-81
SF.REA  = 000010            #5-73
SF.SEC  = 000400            #5-81       103-4494
SF.SER  = 000001            #5-73
SF.SVC  = 000002            #5-73
SF.TER  = 001000            #5-81       103-4498
SF.UNL  = 000040            #5-73
SLTNUM  = ******   GX       *28-1699   *57-2659    66-2989
SPSAV   = ******   GX       *8-220      9-561
SRHFNB    015552 R          #98-4256    103-4513
STAN      000036 R          #6-121     *57-2660   *58-2670    66-2995
STRAD     000332 RG         #7-191      90-3821    90-3844    103-4589   104-4619   104-4648
STRADD    000056 R          #6-126     *77-3445    78-3465    78-3467   *78-3468
STRADR    010242 R          *61-2758   *65-2928    65-2930   #65-2972
STRMAX    010240 R          *61-2759    61-2761    61-2778   *64-2897    65-2912    65-2916    65-2935    65-2952   #65-2971
STRPT     000334 RG         #7-192     *90-3826   *90-3857   *103-4539   103-4570  *103-4571  *103-4599  *104-4630  *104-4676
STRPTR    010236 R          *61-2756   *61-2757    61-2762   *64-2883   *64-2884    65-2908    65-2927   #65-2970
STRSIZ    000204 RG         #7-173      90-3820   *90-3824   *90-3827    90-3842    90-3852   *90-3854   *90-3858   *103-4540
                            *103-4572   103-4590  *103-4600   104-4618  *104-4631   104-4647  *104-4677
SUBPAR  = ******   GX       *8-231     *8-321      8-374      16-970
SWR     = 177570            #5-75
SYSFDB  = ******   GX       *8-286     *8-288     *8-292     *8-294     *8-329     *8-373     *8-402     *8-418     *8-448
                            *8-457     *11-741    *11-742    *11-743    *12-797    *13-834    *14-879    *16-948     17-1086
                            *26-1569   *61-2752   *61-2753   *61-2754   *61-2777   *61-2778   *62-2792   *62-2808   *63-2841
                            *63-2863   *64-2888   *64-2889   *64-2890   *64-2892   *64-2895   *65-2919   *65-2920   *65-2922
                            *65-2923   *65-2926   *65-2956   *65-2959   *65-2960   *65-2961   *67-3017   *67-3027   *67-3037
                            *67-3046   *67-3056   *67-3065   *67-3074   *67-3083   *67-3093   *67-3103   *67-3110   *69-3175
                            *69-3183   *69-3184   *69-3185   *69-3190   *69-3191   *69-3192   *70-3239   *71-3257   *72-3280
                            *73-3303   *74-3318   *75-3344   *75-3372   *76-3392   *76-3408   *79-3506   *80-3542   *86-3686
                            *87-3726   *88-3746   *88-3747   *88-3748   *88-3762   *94-4060   *94-4062   *94-4063   *94-4064
                            *94-4075   *94-4076   *94-4077   *98-4257   *98-4258   *98-4259   *98-4272   *102-4451  *102-4453
                            *102-4454  *102-4455  *102-4466  *102-4467  *102-4468  *103-4524  *103-4526  *103-4530  *103-4536
                            *105-4707
S.ADD     000004            #5-81      *94-4042
S.CIR     000016            #5-81      *94-4048
S.COST    000001            #5-73
S.DEV     000020            #5-81      *94-4050
S.DIA     000046            #5-81       103-4505
S.DPA     000032            #5-81      *94-4056   *94-4057
S.DPC     000036            #5-81      *94-4058
S.DUM     000050            #5-81       103-4509
S.FLAG    000002            #5-81      *94-4041   *103-4531
S.FLG     000000            #5-73      *66-3000   *66-3001   *66-3004    108-4813  *108-4832
```

```
                                                                   F  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50     PAGE 15
```

Table of contents

```
     5-   54    MACRO DEFINITIONS
     6-   92    CONSTANT DEFINITIONS
     7-  121    LOCAL DATA
     8-  198    $CXIO - CEX I/O
     9-  243    $PRIO - PROCESS I/O
    10-  313    CR.MC - CREATE MICROCODE PARTITION
    12-  468    PR.LBL - READ LABEL BLOCK
    13-  580    EXTXPT - EXTEND XPT'S DDB
    14-  616    PR.PAR - FIND PARTITION
    15-  754    PR.SUB - CREATE SUB-PARTITION
    16-  872    PR.XFR - READ PROCESS INTO CORE
    17-  917    FNDFDV - FIND A PDV ADDRESS
```

```
                                        .SBTTL  PR.LBL - READ LABEL BLOCK
                                ;+
                                ; PR.LBL - READ LABEL BLOCK
                                ;
                                ; THIS ROUTINE IS CALLED TO READ THE PROCESS LABEL BLOCK AND TO VERIFY
                                ; THAT THE FILE IS A VALID PROCESS IMAGE. ON EXIT, THE LOGICAL BLOCK I/O
                                ; POINTERS ARE SETUP TO ACCESS THE FIRST TEXT BLOCK.
                                ;
                                ; INPUTS:
                                ;       $NAME= RAD50 PDV NAME
                                ;
                                ; OUTPUTS:
                                ;       C-BIT=SUCCESS/FAILURE
                                ;       R0=LABEL BLOCK BUFFER ADDRESS
                                ;       R1,R2=DESTROYED
                                ;       $LBN=FIRST TEXT BLOCK LOGICAL BLOCK NUMBER
                                ;       $LLEN=PROCESS SIZE IN BYTES
                                ;       BLOX=PROCESS SIZE IN 32. WORD BLOCKS
                                ;-

001002  012700  000000'         PR.LBL: MOV     #DSKBUF,R0        ; GET BUFFER ADDRESS
001006                                  CALL    $RLBL             ; READ LABEL BLOCK
001012  103554                          BCS     101$              ; IF CS, FAILURE
001014  032760  000000G 000000G         BIT     #TS$NHD,L$BFLG(R0) ; DOES PROCESS HAVE A HEADER?
001022  001557                          BEQ     111$              ; IF EQ, YES
001024  016001  000000G                 MOV     L$BLDZ(R0),R1     ; GET LOAD SIZE IN BLOCKS
001030  016002  000000G                 MOV     L$BEXT(R0),R2     ; GET EXTEND SIZE FROM LABEL BLOCK
001034  010167  000260'                 MOV     R1,$LDSIZ         ; SAVE LOAD SIZE FOR LATER

001040  026001  000000G                 CMP     L$BMXZ(R0),R1     ; IS PROCESS OVERLAID?
001044  001146                          BNE     111$              ; IF NE, YES

001046  022767  114224  000000G         CMP     #^RXPT,$NAME      ; LOAD XPT ?
001054  001403                          BEQ     1$                ; IS YES - BRANCH

001056  022701  000200                  CMP     #200,R1           ; IS PROCESS > 4K WORDS ?
001062  103543                          BLO     121$              ; IF LO, YES
001064  032767  000000G 000000G 1$:     BIT     #FL.CEX,$FLAGS    ; IS THIS THE COMM EXEC?
001072  001064                          BNE     20$               ; IF NE, YES
001074  022760  120000  000000G         CMP     #120000,L$BSA(R0) ; WAS PROCESS BUILT FOR APR5?
001102  001137                          BNE     131$              ; IF NE, NO
                                ;
                                ; CALCULATE LINE TABLES SIZE
                                ;
001104  005767  000002G                 TST     $PRINC+2          ; WAS THE LINE TABLE SIZE SPECIFIED ?
001110  001412                          BEQ     5$                ; IF EQ, NO .. ASSUME NO LINE TABLES
001112  010146                          MOV     R1,-(SP)          ; SAVE LOAD SIZE
001114  016701  000000G                 MOV     $$INC,R1          ; GET THE LINE TABLE COUNT
001120  016700  000000G                 MOV     $PRLLN,R0         ; AND THE LINE TABLE SIZE
001124                                  CALL    $MUL              ; GET SIZE IN BYTES OF THE LINE TABLES
001130  010167  000146'                 MOV     R1,EXTEND         ; SAVE THE SIZE OF THE LINE TABLES
001134  012601                          MOV     (SP)+,R1          ; RESTORE THE LOAD SIZE
001136  032767  000000G 000000G 5$:     BIT     #FL.LLC,$FLAGS    ; IS THIS AN LLC PROCESS ?
001144  001016                          BNE     7$                ; IF NE, YES
                                ;
                                ; DDM/DLC PROCESS
                                ;
```

```
                                                                                E  5
VPIO       CREATED BY  MACRO  ON 28-AUG-85 AT 20:04      PAGE 1

SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

BLOX       000142 R         #7-142     *11-459    *12-547    *12-548     14-711     15-793     15-816     15-845
BYTES      000144 R         #7-143     *11-462    *12-553
CR.MC      000220 R          9-298     #10-325
C$LEN      000012            13-605
DSKBUF     000000 R         #7-194      11-454     12-488     12-561
D$END    = 000104            13-599
D$MLL      000040            13-604
D$NN       000060            13-600
ERROR      000250 R         #7-167     *9-267      9-279      9-287     *9-306     *9-308      10-345     10-367     10-377
                            *10-393    *10-398    *12-568    *12-570    *12-572    *12-574    *12-576     14-733    *14-741
                            *14-743    *14-745    *14-747    *14-749    *14-751    *15-866    *15-868    *16-912
ERR10      000012           #6-106      12-570
ERR12      000014           #6-107      12-574
ERR14      000016           #6-108      12-572
ERR16      000020           #6-109      12-576
ERR18      000022           #6-110      14-741
ERR2       000002           #6-102      9-306
ERR20      000024           #6-111      14-743
ERR22      000026           #6-112      14-745
ERR24      000030           #6-113      14-747
ERR26      000032           #6-114      14-749
ERR28      000034           #6-115      14-751
ERR30      000036           #6-116      15-866
ERR32      000040           #6-117      15-868
ERR34      000042           #6-118      16-912
ERR4       000004           #6-103
ERR6       000006           #6-104      9-308
ERR8       000010           #6-105      12-568
EXTEND     000146 R         #7-148     *12-518    *12-531     12-539
EXTXPT     001424 R          12-545    #13-595
E$NBR      000014           #5-70
E$NBS      000020           #5-70
E$NCR      000034           #5-70
E$NCS      000036           #5-70
E$NIC      000044           #5-70
E$NLEN     000050           #5-70
E$NLLA     000012           #5-70
E$NLNK     000000           #5-70
E$NML      000040           #5-70
E$NMR      000024           #5-70
E$NMS      000030           #5-70
E$NNCD     000002           #5-70
E$NRT      000042           #5-70
E$NRTP     000005           #5-70
E$NSEG     000010           #5-70
E$NTIM     000046           #5-70
E$NUSE     000004           #5-70
E$STRT     000006           #5-70
FE.EXV   = 000004            14-707
FE.PLA   = 000020            9-290      10-381
FL.CEX   = ******   GX       12-505     14-664     14-697
FL.LLC   = ******   GX       12-520
```

```
VPRM

SYMB

SYMB

X
Y
$ALP
$ANY
$BLA
$DIG
$DNL
$EOS
$EXI
$FAI
$GOF
$GPR
$LAM
$NUM
$RAD
$RON
$STR
$SUE
$$PR
$$$F
$$$K
```

```
                                                                                F  5
VPIO       CREATED BY  MACRO  ON 28-AUG-85 AT 20:04      PAGE 2
```

```
                                                                     E  6
VPRMPT      CREATED BY  MACRO  ON 29-JUN-85 AT 02:55        PAGE 1

SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL  VALUE                REFERENCES

X          000000 RG         #6-79
Y          000000 RG         #6-79
$ALPHA  =  000022            #6-79
$ANY    =  000020            #6-79
$BLANK  =  000006            #6-79
$DIGIT  =  000024            #6-79
$DNUMB  =  000014            #6-79
$EOS    =  000012            #6-79
$EXIT   =  000000            #6-79
$FAIL   =  177777            #6-79
$GOFF   =  ******             4-1         6-79
$GPRM   =  000001            #5-69        6-79
$LAMDA  =  000000            #6-79
$NUMBR  =  000002            #6-79
$RAD50  =  000016            #6-79
$RONLY  =  000001            #5-73        6-79        6-79        6-79        6-79
$STRNG  =  000004            #6-79
$SUBXP  =  000010            #6-79
$$PRMP     000000 RG         #6-79
$$$FLG  =  177777            #6-79
$$$KEY  =  177777            #6-79
```

```
                                  ;+
                                  ; PRSYM - LOOK FOR SPECIAL PROCESS SYMBOLS
                                  ;
                                  ; INPUTS:
                                  ;         $DFNAM=ASCIZ PROCESS NAME
                                  ;
                                  ; OUTPUTS:
                                  ;         C-BIT=SUCCESS/FAILURE
                                  ;         PVAL=DISPATCH TABLE ADDRESS
                                  ;-
000600  012700  000000G           PRSYM:  MOV     #$DFNAM,R0      ; FORM DISPATCH TABLE ADDRESS NAME
000604  012701  000025'                   MOV     #DSPTAB+1,R1    ; BY STARTING WITH "$",
000610  012702  000004                    MOV     #4,R2           ; APPENDING THE PROCESS NAME,
000614  005302                    10$:    DEC     R2              ; APPENDING "TB",
000616  112021                            MOVB    (R0)+,(R1)+     ; AND BLANK FILLING OUT TO 6 CHARACTERS.
000620  001375                            BNE     10$             ;       ...
000622  112761  000124  177777            MOVB    #'T,-1(R1)      ;       ...
000630  112721  000102                    MOVB    #'B,(R1)+       ;       ...
000634  005702                            TST     R2              ; BLANK FILL IF NEEDED
000636  001404                            BEQ     17$             ;       ...
000640  112721  000040            15$:    MOVB    #' ,(R1)+       ;       ...
000644  005302                            DEC     R2
000646  001374                            BNE     15$             ; ...
000650                            17$:
                                          .IF DF   R$$MPL                                             ;[MP01]

                                          MOV     #$DFNAM,R0      ; FORM DISPATCH TABLE ADDRESS NAME
                                          MOV     #VECTAB+1,R1    ; BY STARTING WITH "$",
                                  20$:    MOVB    (R0)+,(R1)+     ; COPY NAME
                                          BNE     20$             ;       ...
                                          MOVB    #'V,-1(R1)      ; CREATE $XXXVT SYMBOL
                                          MOVB    #'T,(R1)        ;       ...

                                          MOV     #VECTAB,R0      ; NOW TAKE THE ASCII NAME
                                          MOV     PC,R1           ; AND CONVERT TO RAD50
                                          CALL    $CAT5           ; STORING THE RAD50 NAME IN THE GLOBAL
                                          MOV     R1,VTSYM        ; SYMBOL LOOKUP CONTROL BLOCK
                                          MOV     PC,R1           ;       ...
                                          CALL    $CAT5           ;       ...
                                          MOV     R1,VTSYM+2      ;       ...

                                          MOV     #$DFNAM,R0      ; FORM DISPATCH TABLE ADDRESS NAME
                                          MOV     #VECLEN+1,R1    ; BY STARTING WITH "$",
                                  25$:    MOVB    (R0)+,(R1)+     ;       ...
                                          BNE     25$             ;       ...
                                          MOVB    #'V,-1(R1)      ; CREATE $XXXVL SYMBOL
                                          MOVB    #'L,(R1)        ;       ...

                                          MOV     #VECLEN,R0      ; NOW TAKE THE ASCII NAME
                                          MOV     PC,R1           ; AND CONVERT TO RAD50
                                          CALL    $CAT5           ; STORING THE RAD50 NAME IN THE GLOBAL
                                          MOV     R1,VLSYM        ; SYMBOL LOOKUP CONTROL BLOCK
                                          MOV     PC,R1           ;       ...
                                          CALL    $CAT5           ;       ...
                                          MOV     R1,VLSYM+2      ;       ...

                                          .ENDC                                                       ;[MP01]
```

```
003640  010467  000050G                MOV     R4,N$MHC2+4+$BFR         ; STORE VIRTUAL ADDR OF START OF MATRIX
003644  000406                         BR      30$                      ; CONTINUE

003646  005067  000044G       20$:     CLR     N$MHC2+$BFR              ; INIT AREA MINHOP/MINCOST ENTRIES
003652  005067  000046G                CLR     N$MHC2+2+$BFR            ; ...
003656  005067  000050G                CLR     N$MHC2+4+$BFR            ; ...

003662                        30$:     PUTRC                            ; REWRITE UPDATED DDB

003666  005046                         CLR     -(SP)                    ; INIT STACK AS FLAG INDICATING
                                                                        ; AREA OR NODE MTRIX (0=NODE,-1=AREA)
003670  016700  000036G                MOV     N$MHC1+$BFR,R0           ; COPY NUMBER OF ENTRIES
003674  016701  000040G                MOV     N$MHC1+2+$BFR,R1         ; COPY BIAS OF NODE MIN HOP/COST
003700  016702  000042G                MOV     N$MHC1+4+$BFR,R2         ; COPY VIRTUAL ADDRESS
003704  006300                40$:     ASL     R0                       ; CONVERT TO BYTES
003706  010004                         MOV     R0,R4                    ; SAVE SIZE

003710  022704  001000        50$:     CMP     #512.,R4                 ; SIZE > ONE TRANSFER
003714  002002                         BGE     60$                      ; IF NO - BRANCH
003716  012704  001000                 MOV     #512.,R4                 ; IF YES - TRANSFER ONE BLOCK
003722  012703  000000G       60$:     MOV     #$BFR,R3                 ; GET BUFFER ADDRESS
003726  010405                         MOV     R4,R5                    ; SAVE SIZE

003730  105023                70$:     CLRB    (R3)+                    ; INIT BUFFER
003732                                 SOB     R5,70$                   ; IF MORE -BRANCH

003736                                 PUTAD   R2,R4,R1                 ; WRITE TO IMAGE
003756  062702  001000                 ADD     #512.,R2                 ; ADJUST VIRTUAL ADDRESS
003762  160400                         SUB     R4,R0                    ; ADJUST REMAINING SIZE
003764  010004                         MOV     R0,R4                    ; SET NEW REMAINING SIZE
003766  001350                         BNE     50$                      ; IF MORE - BRANCH

003770  005716                         TST     (SP)                     ; CHECK FLAG
003772  100424                         BMI     80$                      ; IF AREA HOP/COST CLEARED - BRANCH
                                                                        ; IF NOT - INIT AREA MIN HOP/COST

003774  016702  000000G                MOV     $PDVA,R2                 ; COPY XPT PDV ADRESS
004000  016202  000000G                MOV     Z.DAT(R2),R2             ; GET XPT DDB ADDRESS
004004                                 GETRV   R2,#N$XLEN               ; READ DDB
004022  016700  000044G                MOV     N$MHC2+$BFR,R0           ; COPY # OF AREA MINHOP/MINCOST ENTRIES
004026  001406                         BEQ     80$                      ; IF NONE - BRANCH
004030  016701  000046G                MOV     N$MHC2+2+$BFR,R1         ; GET AREA MIN HOP/COST BIAS
004034  016702  000050G                MOV     N$MHC2+4+$BFR,R2         ; GET AREA MIN HOP/COST ADDRESS
004040  005316                         DEC     (SP)                     ; SET FLAG INDICATING AREA HOP/COST
004042  000720                         BR      40$                      ; INIT AREA MIN HOP/COST MATRIX

004044  005726                80$:     TST     (SP)+                    ; CLEAN UP STACK
004046                        90$:     RETURN


004050                        100$:    EMSG$R  69                       ; ELSE ALLOCATION FAILURE

```

E 9

```
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05     PAGE 6
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE            REFERENCES

$CMFRK   = ******   GX    *9-384     *9-392     *9-393
$CTSYM     004504 RG      #19-1196
$DECPT   = ******   GX     12-618     13-707     14-817     16-989     17-1096
$DFN     = ******   GX     7-252
$DFNAM   = ******   GX     10-413
$ERRQX   = ******   GX     7-321
$ERR44   = ******   GX     6-206
$ERR45   = ******   GX     6-207
$ERR46   = ******   GX     6-208
$ERR47   = ******   GX     6-209
$ERR49   = ******   GX     10-504
$ERR50   = ******   GX     10-505
$ERR51   = ******   GX     6-214
$ERR53   = ******   GX     6-216
$ERR54   = ******   GX     6-217
$ERR55   = ******   GX     6-218
$ERR56   = ******   GX     6-219
$ERR57   = ******   GX     6-220
$ERR58   = ******   GX     6-221
$ERR59   = ******   GX     6-222
$ERR60   = ******   GX     6-223
$ERR61   = ******   GX     6-224
$ERR62   = ******   GX     6-225
$ERR63   = ******   GX     6-226
$ERR64   = ******   GX     6-227
$ERR65   = ******   GX     6-228
$ERR66   = ******   GX     6-229
$ERR67   = ******   GX     6-230
$ERR69   = ******   GX     16-1077
$ERR70   = ******   GX     11-598     13-795     17-1147
$ERR9D   = ******   GX     14-851     15-966
$ERR99   = ******   GX     12-689
$GETAD   = ******   GX     7-308      11-577     14-838     15-951     17-1123
$GETRV   = ******   GX     7-270      11-571     12-618     12-640     13-707     13-720     13-788     14-817     14-835
                           14-845     15-877     15-928     15-942     16-989     16-1010    16-1065    17-1096    17-1111
                           17-1141
$LDSIZ   = ******   GX     7-271
$LLCTA   = ******   GX     11-542
$MUL     = ******   GX     13-715     14-825     15-933     17-1102
$NAME    = ******   GX     7-251      9-375
$NTLHB   = ******   GX     8-340     *8-341      10-487
$PDVA    = ******   GX     7-248      7-268      8-346      12-638     13-717     14-813     15-873     15-920     16-1008
                           16-1063    17-1109
$PDVNM   = ******   GX     18-1165
$PDVTA   = ******   GX     11-554     18-1164    18-1177
$PRCLN     000112 RG      #6-165
$PREXT     000102 RG       6-156     #6-163
$PRINC     000070 RG       6-134      6-148     #6-156
$PRIO    = ******   GX     7-259
$PRLLN     000100 RG      #6-158
$PUTAD   = ******   GX     7-311      8-351      11-590     12-681     13-750     13-774     14-840     15-899     15-917
                           15-958     16-1053    17-1132    17-1138
```

F 9

```
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05     PAGE 7
```

```
                                                    .SBTTL  $AZFS - SET UP ASCIZ FILE SPEC
                                        ;+
                                        ; $AZFS - SETUP ASCIZ FILE SPEC
                                        ;
                                        ; INPUTS:
                                        ;           R0=ASCIZ UIC ADDRESS (0=USE DEFAULT)
                                        ;           R1=ASCIZ FILE NAME ADDRESS (0=USE DEFAULT)
                                        ;           R2=ASCIZ EXTENSION ADDRESS (0=USE DEFAULT)
                                        ;
                                        ; OUTPUTS:
                                        ;           R1=ASCIZ FILE SPEC ADDRESS
                                        ;           R2=FILE SPEC LENGTH (EXCLUDING ZERO BYTE)
                                        ;-

 001042  010246                         $AZFS:: MOV     R2,-(SP)                ; SAVE EXTENSION ADDRESS
 001044  012702  001374'                        MOV     #FILE,R2                ; PLACE WHERE FILE SPEC WILL BE ASSEMBLED
 001050  005700                                 TST     R0                      ; IS THERE A UIC ADDRESS?
 001052  001002                                 BNE     10$                     ; IF NE, YES
 001054  012700  001534'                        MOV     #$DFUIC,R0              ; ELSE USE DEFAULT UIC
 001060  112022                         10$:    MOVB    (R0)+,(R2)+             ; COPY ASCIZ UIC INTO FILE SPEC
 001062  001376                                 BNE     10$                     ; ...
 001064  005302                                 DEC     R2                      ; DISCARD ZERO BYTE
 001066  010100                                 MOV     R1,R0                   ; COPY FILE NAME ADDRESS
 001070  001002                                 BNE     15$                     ; IF NE, THERE IS ONE
 001072  012700  001552'                        MOV     #$DFNAM,R0              ; ELSE USE DEFAULT NAME
 001076  010001                         15$:    MOV     R0,R1                   ; SAVE FILE NAME ADDRESS FOR LATER
 001100  112022                         20$:    MOVB    (R0)+,(R2)+             ; COPY ASCIZ FILE NAME INTO FILE SPEC
 001102  001376                                 BNE     20$                     ; ...
 001104  112762  000056  177777                 MOVB    #'.,-1(R2)              ; DEPOSIT "."
 001112  012600                                 MOV     (SP)+,R0                ; RESTORE EXTENSION ADDRESS
 001114  001002                                 BNE     30$                     ; IF NE, THERE IS ONE
 001116  012700  001606'                        MOV     #DFEXT,R0               ; ELSE USE DEFAULT EXTENSION
 001122  112022                         30$:    MOVB    (R0)+,(R2)+             ; COPY ASCIZ EXTENSION INTO FILE SPEC
 001124  001376                                 BNE     30$                     ; ...
 001126  005302                                 DEC     R2                      ; DISCARD ZERO BYTE
 001130  010267  000226                         MOV     R2,FNPTR                ; SAVE POINTER FOR POSSIBLE ERROR
 001134  012700  000001                         MOV     #1,R0                   ; SETUP LOCAL FILE NAME POINTER
 001140                                         CALL    SLFNP                   ; FOR NEXT GCL LEVEL
 001144  112120                         40$:    MOVB    (R1)+,(R0)+             ; COPY FILE NAME INTO LOCAL AREA
 001146  001376                                 BNE     40$                     ; ...
 001150  012701  001374'                        MOV     #FILE,R1                ; SET FILE SPEC ADDRESS
 001154  160102                                 SUB     R1,R2                   ; SET FILE SPEC LENGTH
 001156                                         RETURN
```

```
    756                                            .SBTTL  CEX DATA
    757                                    ;
    758                                    ; IMAGES MAINTAINED BY MODULES VIMI AND VIMO
    759                                    ;
    760
    761 003704  000000                     $PDVTA::.WORD   0                          ; ADDRESS OF PDV ADDRESS TABLE
    762 003706  000000                     $SLTMA::.WORD   0                          ; ADDRESS OF SLT ADDRESS TABLE
    763 003710  000000                     $LLCTA::.WORD   0                          ; ADDRESS OF LLC REVERSE MAPPING TABLE
    764 003712  000000                     $PDVNM::.WORD   0                          ; NUMBER OF PDV ENTRIES
    765 003714  000000                     $SLTNM::.WORD   0                          ; NUMBER OF SYSTEM LINES
    766 003716  000000                     $CCBNM::.WORD   0                          ; # OF CCB'S ALLOCATED
    767 003720  000000                     $CCBSZ::.WORD   0                          ; # BYTES / CCB
    768 003722  000000                     $RDBNM::.WORD   0                          ; # OF RDB'S ALLOCATED
    769 003724  000000                     $RDBSZ::.WORD   0                          ; # BYTES / RDB
    770 003726  000000                     $SDBNM::.WORD   0                          ; # OF SDB'S ALLOCATED
    771 003730  000000                     $SDBSZ::.WORD   0                          ; # BYTES / SDB
    772 003732  000000                     $CCBCT::.WORD   0                          ; COUNTER OF CURRENT # OF CCB'S
    773 003734  000000                     $CCBAF::.WORD   0                          ; COUNTER OF # OF CCB ALLOCATION FAILURES
    774 003736  000000                     $LDBAF::.WORD   0                          ; COUNTER OF # OF LDB ALLOCATION FAILURES
    775 003740  000000                     $CCBAL::.WORD   0                          ; COUNTER OF # OF DYNAMIC CCB ALLOCATIONS
    776 003742  000000                     $RDBTH::.WORD   0                          ; POOL THRESHOLD FOR LARGE DATA BUFFER ALLOCATION FA
    777 003744  000000                     $CMPDV::.WORD   0                          ; PDV INDEX OF CURRENT DISPATCHED PROCESS
    778 003746  000000                     $ZTIME::.WORD   0                          ; TIME SINCE SYSTEM LAST ZEROED
    779 003750  000000                             .WORD   0                          ; TIME SINCE COUNTERS LAST ZEROED
    780
    781                                            .IF DF  M$$PRO                                                     ;[MP01]
    782                                            .WORD   0                          ; UNINUS RUN MASK                  ;[MP01]
    783                                            .ENDC                                                              ;[MP01]
    784
    785 003752  000000                     $CMFRK::.WORD   0                          ; COMM EXEC FORK BLOCK
    786 003754  000000                             .WORD   0                          ;       ...
    787 003756  000000                             .WORD   0                          ;       ...
    788 003760  000000                             .WORD   0                          ;       ...
    789 003762  000000                             .WORD   0                          ; APR5 BIAS (FREE WORD ON UNMAPPED SYSTEMS)
    790 003764                             $T100Q::                                   ; 100 MSEC TIMER QUEUE
    791 003764  000000                     $T50Q::  .WORD  0                          ;  50 MSEC TIMER QUEUE (TO BE DELETED)
    792 003766  000000                             .WORD   0                          ;  ...
    793 003770  000000                     $STMFC::.WORD   0                          ; FUNCTION CODE TO DISPATCH 50 MSEC TIMEOUTNS
    794 003772                             $T100C::                                   ; 100 MSEC CLOCK QUEUE ENTRY
    795 003772  000000  000000  000000     $T50CL::.WORD   0,0,0,0,0                  ; 50 MSEC CLOCK QUEUE ENTRY
    796 004004  000000                             .WORD   0                          ; ADDRESS OF PROCESSING ROUTINE
    797 004006  000000                             .WORD   0                          ; APR5 BIAS ON MAPPED SYSTEMS (FREE ON UNMAPPED)
    798                                            .IF DF  M$$PRO                                                     ;[MP01]
    799                                            .WORD   0                          ; UNIBUS RUN MASK                  ;[MP01]
    800                                            .ENDC                                                              ;[MP01]
    801
    802 004010  000000                     $LTMFC::.WORD   0                          ; FUNCTION CODE TO DISPATCH LONG TIMEOUTS
    803 004012  000000  000000  000000     $T1SCL::.WORD   0,0,0,0,0                  ; 1 SECOND CLOCK QUEUE ENTRY
    804 004024  000000                             .WORD   0                          ; ADDRESS OF PROCESSING ROUTINE
    805 004026  000000                             .WORD   0                          ; APR5 BIAS ON MAPPED SYSTEMS (FREE ON UNMAPPED)
    806
    807                                            .IF DF  M$$PRO                                                     ;[MP01]
    808                                            .WORD   0                          ; UNIBUS RUN MASK                  ;[MP01]
    809                                            .ENDC                                                              ;[MP01]
    810
    811
    812 004030  000000                     $DECPT::.WORD   0                          ; DECNET HOME BLOCK POINTER
```

```
                                                        .SBTTL  $CEACC - ACCESS BLOCK IN EXTENDED POOL

                                                ;+
                                                ; **-$CEACC-ACCESS BLOCK IN EXTENDED POOL
                                                ;
                                                ; THIS SUBROUTINE MAPS TO A BLOCK IN EXTENDED SINGLE WORD POOL,
                                                ; AND RETURNS A VIRTUAL ADDRESS TO ACCESS THE BLOCK.
                                                ;
                                                ; INPUTS:
                                                ;       2(SP)  - ADDRESS OF BLOCK TO BE ACCESSED
                                                ;
                                                ; OUTPUTS:
                                                ;       2(SP)  - VIRTUAL ADDRESS OF MAPPED BLOCK
                                                ;       $RBLCK - APR BIAS OF BLOCK TO BE ACCESSED (0 IF FROM DSR)
                                                ;
                                                ;-

005622                                          $CEACC::
005622  005067  172632                                  CLR     $RBLCK          ; ASSUME BLOCK IS IN DSR
005626  032766  000001  000002                          BIT     #1,2(SP)        ; IS THE BLOCK IN EXTENDED POOL ?
005634  001427                                          BEQ     10$             ; IF EQ, NO - ALREADY MAPPED TO IT
005636                                                  SAVRG   <R0>            ; ELSE, SAVE R0
005640  016600  000004                                  MOV     4(SP),R0        ; COPY BLOCK ADDRESS
005644  042766  177701  000004                          BIC     #177701,4(SP)   ; RETURN VIRTUAL ADDRESS
005652  052766  140000  000004                          BIS     #140000,4(SP)   ; ... MAPPED THROUGH KISAR6
005660                                                  ASR$    6,R0            ; CONVERT TO BLOCK OFFSET
005674  042700  176000                                  BIC     #176000,R0      ; CLEAR EXTRANEOUS BITS
005700  016767  176200  172552                          MOV     $XBIAS,$RBLCK   ; MAP TO BASE ADDRESS OF POOL
005706  060067  172546                                  ADD     R0,$RBLCK       ; ... UPDATE TO PROPER BLOCK
005712                                                  RESRG   <R0>            ; RESTORE R0
005714                                          10$:    RETURN                  ; RETURN TO THE CALLER
```

```
SYMBOL CROSS REFERENCE                                            CREF   04.00

SYMBOL  VALUE               REFERENCES

PF.XDF  = 000001   G        #15-623
PF.XGA  = 000004   G        #15-625
PREFIX    001622 R          #14-409      14-432
PREF2     001631 R          #14-410     *14-424     *14-427
PSTART    001002 RG         #7-190
PUMR      001010 RG         #7-193
QF$ACT  = 000001            #5-80
QF$ALL  = 000002            #5-80
QF$KNO  = 000004            #5-80
QF$LOO  = 000010            #5-80
Q.IDAE  = 000012   G         6-102
Q.IOFN  = 000002   G         6-96        6-102       6-102
Q.IOPL  = 000014   G         6-102      *14-418     *14-419
Q.IOSB  = 000010   G         6-102       6-102
RA        000730 RC         #7-172
RB        000732 RG         #7-173
RQB       003430 RG         #15-635      15-636
RQBST   = 003430 RG         #15-636
R$$EIS  = ******             24-1320
R$$MPL  = ******             7-151       15-463      15-502      15-591      15-612      16-657      16-733      16-746      17-833
                             17-870      17-944      18-1002     19-1059     19-1085     19-1103     20-1158     21-1192     26-1361
R$$11D  = ******             5-79        24-1320
R$$11M  = 000000             5-79
R$$11S  = ******             5-79
SBFLG     000772 RG         #7-186
SBNAM     000766 RG         #7-184
SBPAR     000762 RG         #7-182
SBPCB     000746 RG         #7-177
SBPOOL    000760 RG         #7-181
SF.ACT  = 000200   G        #5-78
SF.ENA  = 000100   G        #5-78
SF.LPB  = 000004   G        #5-78
SF.MFL  = 000040   G        #5-78
SF.PAC  = 000020   G        #5-78
SF.REA  = 000010   G        #5-78
SF.SER  = 000001   G        #5-78
SF.SVC  = 000002   G        #5-78
SF.UNL  = 000040   G        #5-78
SLFNP     001332 R           8-255       10-318     #12-369
SLTNUM    001040 RG         #7-216
SPARE1    004220 RG         #17-916
SPARE2    004224 RG         #17-917
SPSAV     001006 RG         #7-192
SUBPAR    000744 RG         #7-176
SYSDNB    000416 R           6-113      #6-114
SYSFDB    000256 RG         #6-111       6-117       19-1082     21-1202     23-1287
S.BFHD  = 000020             6-89        6-89
S.COST    000001   G        #5-78
S.ETEF  = 000002             6-102
S.FDB   = 000140             6-102       6-102       6-111       6-111
S.FLG     000000   G        #5-78
S.FNAM  = 000006             6-102       6-102       6-114       6-114
```

```
                                   .SBTTL  MACRO DEFINITIONS
                           ;
                           ; MACROS
                           ;
                                   .MCALL  ASL$,EMSG$,VNPDF$,NTLER$,SWTCHI,GETRV,PUTRC,EMSG$R
                                   .MCALL  DCBDF$,UCBDF$,SWTCHO,TCBDF$,CALLR,SAVRG,RESRG,GETAD
                                   .ENABL  LC                      ; ENABLE LOWER CASE

       000000                      VNPDF$                          ; VNP REQUEST BLOCK OFFSETS
       000000                      DCBDF$                          ; DEVICE CONTROL BLOCK OFFSETS
       000000                      UCBDF$                          ; UNIT CONTROL BLOCK OFFSETS
       000000                      TCBDF$                          ; TASK CONTROL BLOCK OFFSETS
```

```
                                                .SBTTL  CHKPP - CHECK PROCESS PARMS
                                        ;+
                                        ; CHKPP - CHECK PROCESS PARAMETERS
                                        ;
                                        ; INPUTS:
                                        ;       LS.OPT SET IF ANY OPTIONAL PARAMETERS
                                        ;       R3=REQUEST BLOCK POINTER
                                        ;               .BLKW   1       ; LO.LIN - NUMBER OF LINE TABLES
                                        ;               .BLKW   1       ; LO.CON - NUMBER OT CONTROLLER TABLES
                                        ;               .BLKW   2       ; LO.PAR - PARTITION NAME
                                        ;
                                        ; OUTPUTS:
                                        ;       R3=DESTROYED
                                        ;       $$INC=INCREMENT SIZE IN 32. WORD BLOCKS
                                        ;       $$PAR=PARTITION NAME
                                        ;-
000552                                  CHKPP:
000552  012767  000000C 000000G                 MOV     #MS.INC!MS.EXT,$MISS ; ASSUME PROCESS EXTENSION FIELDS ABSENT
000560  032767  000400  000004G                 BIT     #LS.OPT,$$LRB+4 ; ANY OPTIONAL PARAMETERS INCLUDED?
000566  001002                                  BNE     10$             ; IF NE, YES
000570  012703  000234'                         MOV     #NULL,R3        ; IF NOT, USE NULL PARAMETERS
000574  012367  000000G                 10$:    MOV     (R3)+,$$INC     ; SET INCREMENT VALUE
000600  001403                                  BEQ     20$             ; IF EQ, LINE TABLE COUNT NOT PRESENT
000602  042767  000000G 000000G                 BIC     #MS.INC,$MISS   ; ELSE, NO LONGER MISSING
000610  005067  000000G                 20$:    CLR     $$EXT           ; CLEAR LLC PROCESS EXTENSION
000614  032767  000000G 000000G                 BIT     #FL.LLC,$FLAGS  ; IS THIS AN LLC ?
000622  001006                                  BNE     30$             ; IF NE, YES
000624  011367  000000G                         MOV     (R3),$$EXT      ; SET CONTROLLER TABLE COUNT
000630  001403                                  BEQ     30$             ; IF EQ, NOT PRESENT
000632  042767  000000G 000000G                 BIC     #MS.EXT,$MISS   ; ELSE, NO LONGER MISSING
000640  005723                          30$:    TST     (R3)+           ; SKIP CONTROLLER TABLE COUNT
000642  012367  000000G                         MOV     (R3)+,$$PAR     ; SET PARTITION NAME
000646  011367  000002G                         MOV     (R3),$$PAR+2    ;       ...
000652  005067  000000G                         CLR     $$DEV           ; INDICATE NO DEVICE NAME (YET)
000656  005067  000000G                         CLR     $$TBL           ; AND NO DISPATCH TABLE ADDRESS
000662  005067  000000G                         CLR     $$LINX          ; ZERO LOGICAL LINK COUNT
000666                                          RETURN
```

```
                                        .SBTTL  CLIN - CHECK CLEAR LINE PARMS
                                ;+
                                ; CLIN - CHECK "CLEAR LINE" REQUEST BLOCK
                                ;
                                ; INPUTS:
                                ;       $SLTA=SLT ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       NONE
                                ;-

002472  016700  000000G         CLIN:   MOV     $SLTA,R0                ; GET SLT ADDRESS
002476  062700  000000G                 ADD     #L.FLG,R0               ; POINT AT STATUS WORD
002502  032710  000000G                 BIT     #LF.RDY,(R0)            ; IS LINE READY?
002506  001002                          BNE     10$                     ; IF NE, YES
002510  005004                          CLR     R4                      ; INDICATE NO WORK TO DO
002512  000405                          BR      20$
002514  032710  000000G         10$:    BIT     #LF.ACT,(R0)            ; IS LINE ACTIVE?
002520  001003                          BNE     101$                    ; IF NE, YES - NOT ALLOWED
002522  052710  000000G                 BIS     #LF.UNL,(R0)            ; INDICATE LINE MARKED FOR UNLOAD
002526                          20$:    RETURN

                                ;
                                ; ERROR CONDITION
                                ;
002530                          101$:   EMSG$R  Z6                      ; LINE STILL ACTIVE
```

```
                                                         F  1
VNTI      CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 2

SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE           REFERENCES

DFSIZ   = 000010         50-2541  #53-2593
DFTBL     006404 R       50-2540  #53-2585   53-2593
DVSZ    = 000010  G     #7-180     103-4544
D$AMXC    000072        *76-3394
D$AMXH    000074        *75-3374
D$BRTM    000100        *76-3418
D$DELF    000045        *67-3066
D$DELW    000046        *67-3075
D$END   = 000104         8-457     61-2753   62-2792   62-2808   63-2841   63-2863   64-2889   65-2920   65-2960
                         67-3017   67-3027   67-3037   67-3046   67-3056   67-3065   67-3074   67-3083   67-3093
                         67-3103   67-3110   69-3184   69-3191   70-3239   71-3257   72-3280   73-3303   74-3318
                         75-3344   75-3372   76-3392   76-3408   88-3747   94-4063   94-4076   98-4258   102-4454
                         102-4467
D$FNB     000034         98-4261   102-4457
D$HOST    000022        *63-2842  *63-2865
D$INAC    000044        *67-3057
D$INCT    000042        *67-3038
D$IPL     000051        *67-3084
D$LID     000020        *64-2893   65-2921   65-2954
D$LNAM    000006         61-2758
D$LNUM    000014        *62-2796  *62-2818
D$LST     000047        *67-3094
D$MAXC    000064        *71-3259
D$MAXH    000066        *72-3282
D$MAXV    000070        *72-3283  *72-3284
D$MLL     000040        *67-3018
D$MNOD    000041        *67-3028
D$NA      000062        *74-3328
D$NBEA    000056        *75-3355
D$NBRA    000054        *8-458
D$NN      000060        *70-3241   74-3320   75-3346   76-3410
D$OUTT    000043        *67-3047
D$RETF    000050        *67-3104
D$RNN     000002         69-3187   88-3750
D$RTMR    000076        *73-3304
D$SEG     000036        *67-3111
D$SER     000032         94-4066
ECLLK     004610 R      #25-1502
ECLLLN    000070 R      #6-135     24-1428  *25-1502
ECLLNK    010340 R      #67-3015
ECLNOD    000072 R      #6-136     24-1441  *25-1505
ENDHEX    000204 RG     #7-171     95-4099
ENDPAS    000132 RG     #7-160     95-4126
ERRTAB    002036 R       9-550     10-591   #10-629
ERR1      000336 R      #10-629    10-629    10-629
ERR10     000742 R      #10-636    10-636    10-636
ERR11     001002 R      #10-637    10-637    10-637
ERR12     001050 R      #10-638    10-638    10-638
ERR13     001116 R      #10-639    10-639    10-639
ERR14     001154 R      #10-640    10-640    10-640
ERR15     001216 R      #10-641    10-641    10-641
ERR16     001252 R      #10-642    10-642    10-642
```

```
                                                         G  1
VNTI      CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 3
```

```
                                                                    F  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 15
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE             REFERENCES

S.HAD      000010          #5-81       *94-4045    *94-4046    *94-4047
S.HST      000006          #5-81       *94-4043
S.LEN      000004          #5-73        108-4834
S.LENG     000052          #5-81        94-4025     94-4036     94-4060     103-4530    103-4536
S.LNK      000000          #5-81
S.LOA      000044          #5-81        103-4501
S.NMST     000002          #5-73
S.OWNR     000003          #5-73
S.PSS      000022          #5-81       *94-4051    *94-4052    *94-4054    *94-4055
S.SEC      000040          #5-81        103-4493
S.TER      000042          #5-81        103-4497
TOPFLG   = ******   GX     *17-1041     17-1074    *17-1100     26-1588     26-1603    *107-4781    107-4782   *108-4819    108-4820
                           *109-4853    109-4854
TPS      = 177564          #5-75
TYPE       000063 R        #6-130      *82-3580     86-3679     87-3727
TY$CHA   = 000400          #5-78
TY$EVE   = 001000          #5-78
TY$STA   = 002000          #5-78
TY$SUM   = 000000          #5-78
T$FLAG     000044          #5-69
T$LIF      000013          #5-69
T$LIFL     000013          #5-69
T$LIFO     000013          #5-69
T$LIFS     000013          #5-69
T$LIN      000000          #5-69
T$LIPS     000006          #5-69
T$LLD      000012          #5-69
T$LLDC     000045          #5-69
T$LLDL     000012          #5-69
T$LLDO     000012          #5-69
T$LLDS     000012          #5-69
T$LLEN     000046          #5-69        24-1416
T$LOPR     000002          #5-69
T$LTCL     000024          #5-69
T$LTIM     000026          #5-69
T$LTPR     000014          #5-69
T$LTPS     000020          #5-69
T$NAPL     000004          #5-69
T$NFE      000000          #5-69
T$NLEN     000010          #5-69
T$NNUL     000002          #5-69
T$NOPL     000006          #5-69
T$NRNI     000042          #5-69
T$NRPL     000005          #5-69
T$NRUL     000007          #5-69
T$NVR      000001          #5-69
T$RPRI     000040          #5-69
T$SVC      000034          #5-69
T$T5       000030          #5-69
T$T6       000032          #5-69
UBMPR    = 170200          #5-75        11-733
UICBUF     000206 RG       #7-174       7-175       101-4401    101-4402    103-4547   *104-4621    104-4622   *104-4624   *104-4652
```

```
                                                                    G  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 16
```

```
                .TITLE  VPIO - VNP PROCESS I/O ROUTINES
                .IDENT  /V05.00/

        ;
        ; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
        ;
        ;
        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
        ; TRANSFERRED.
        ;
        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
        ; CORPORATION.
        ;
        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
        ;
        ; MODULE DESCRIPTION:
        ;
        ;       VNP - COMMON "SET CEX"/"SET PROCESS" I/O ROUTINES
        ;
        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
        ;
        ; IDENT HISTORY:
        ;
        ; 1.00  27-FEB-78
        ;       VERSION 2.0 RELEASE
        ;
        ; 2.00  14-DEC-79
        ;       DECNET-11M/S V3.0
        ;       DECNET-11M-PLUS V1.0
        ;
        ; 3.00  16-APR-82
        ;       DECNET-11M V3.1
        ;       DECNET-11M-PLUS V1.1
        ;
        ; 4.00  07-NOV-83
        ;       DECNET-11M V4.0
        ;       DECNET-11M-PLUS V2.0
        ;
        ; 5.00  22-JUL-85
        ;       DECnet-11M/S V4.2
        ;       DECnet-11M-Plus V3.0
        ;       DECnet-Micro/RSX V1.0
        ;
        ;
```

```
    525 001146  005767  000002G                         TST     $PREXT+2           ; WAS THE CONTROLLER SIZE SPECIFIED ?
    526 001152  001415                                  BEQ     10$                ; IF EQ, NO .. ASSUME NO CONTROLLERS
    527 001154  010146                                  MOV     R1,-(SP)           ; SAVE LOAD SIZE
    528 001156  016701  000000G                         MOV     $$EXT,R1           ; GET THE CONTROLLER TABLE COUNT
    529 001162  016700  000000G                         MOV     $PRCLN,R0          ; AND THE CONTROLLER TABLE SIZE
    530 001166                                          CALL    $MUL               ; CALCULATE SIZE OF CONTROLLER TABLES
    531 001172  060167  000146'                         ADD     R1,EXTEND          ; ADD CONTROLLER TABLE SIZE TO EXTEND SIZE
    532 001176  012601                                  MOV     (SP)+,R1           ; RESTORE THE LOAD SIZE
    533 001200  000402                                  BR      10$                ; AND SKIP LLC PROCESS EXTENSION
    534                                         ;
    535                                         ; LLC PROCESS
    536                                         ;
    537 001202  066702  000000G                 7$:     ADD     $$EXT,R2           ; ADD IN THE LLC DEFINITION EXTEND SIZE
    538
    539 001206  016700  000146'                 10$:    MOV     EXTEND,R0          ; GET THE PROCESS EXTEND SIZE
    540 001212  062700  000077                          ADD     #77,RC             ; ROUND UP TO NEXT BLOCK
    541 001216  042700  000077                          BIC     #77,R0             ; ...
    542 001222                                          ASR$    6,R0               ; CONVERT BYTES TO BLOCKS
    543 001236  060002                                  ADD     R0,R2              ; INCREASE THE EXTEND SIZE
    544
    545 001240                                          CALL    EXTXPT             ; CHECK FOR XPT AND EXTEND IF SO
    546
    547 001244  010167  000142'                 20$:    MOV     R1,BLOX            ; SET PROCESS SIZE IN BLOCKS
    548 001250  060267  000142'                         ADD     R2,BLOX            ;       ...
    549 001254                                          ASL$    6,R1               ; CONVERT FROM BLOCKS TO BYTES
    550 001270                                          ASL$    6,R2               ; BY SHIFTING LEFT 6 PLACES
    551 001304  010167  000000G                         MOV     R1,$LLEN           ; SET TRANSFER SIZE IN BYTES
    552 001310  060102                                  ADD     R1,R2              ; ADD LOAD SIZE TO EXTEND SIZE
    553 001312  010267  000144'                         MOV     R2,BYTES           ; SET PROCESS SIZE IN BYTES
    554
    555 001316  022767  114224  000000G                 CMP     #^RXPT,$NAME       ; ALLOW XPT > 4KW
    556 001324  001403                                  BEQ     25$                ; IF XPT - BRANCH
    557
    558 001326  022702  020000                          CMP     #20000,R2          ; IS PROCESS + EXTENSION > 4K WORDS?
    559 001332  103427                                  BLO     141$               ; IF LO, YES
    560 001334  000241                          25$:    CLC                        ; INDICATE SUCCESS
    561 001336  012700  000000'                         MOV     #DSKBUF,R0         ; RESTORE ADDRESS OF LABEL BLOCK BUFFER
    562 001342                                          RETURN
    563
    564                                         ;
    565                                         ; ERROR CONDITIONS
    566                                         ;
    567 001344  116777  000000G 000000G 101$:   MOVB    $IOSB,@$CLIOS      ; COPY ERROR CODE
    568 001352  012767  000010  000250'         MOV     #ERR8,ERROR        ; LABEL BLOCK READ FAILURE
    569 001360  000417                                  BR      142$
    570 001362  012767  000012  000250' 111$:   MOV     #ERR10,ERROR       ; INVALID PROCESS IMAGE
    571 001370  000413                                  BR      142$
    572 001372  012767  000016  000250' 121$:   MOV     #ERR14,ERROR       ; PROCESS EXCEEDS 4K WORDS
    573 001400  000407                                  BR      142$
    574 001402  012767  000014  000250' 131$:   MOV     #ERR12,ERROR       ; PROCESS NOT BUILT FOR APR5
    575 001410  000403                                  BR      142$
    576 001412  012767  000020  000250' 141$:   MOV     #ERR16,ERROR       ; PROCESS EXTENSION EXCEEDS 4K WORDS
    577 001420  000261                  142$:   SEC                        ; INDICATE FAILURE
    578 001422                                          RETURN
```

```
                                                                  F  5
VPIO       CREATED BY  MACRO  ON 28-AUG-85 AT 20:04     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL   VALUE             REFERENCES

FL.LMC  = ******   GX    10-343
FL.TOP  = ******   GX    14-632     14-668     14-693     15-823     15-847
FNDPDV    003100 R      #17-930
F.HIBK  = ******   GX    14-681
F.RSIZ  = ******   GX   *9-292     *10-333    *10-383    *13-599    *14-650    *15-805    *16-901
I$$AS   = ******         12-542     13-610
KDPMC     000166 R      #7-157      11-431
KDZMC     000200 R      #7-158      11-433
L$BEXT  = ******   GX    12-494
L$BFLG  = ******   GX    12-491
L$BLDZ  = ******   GX    11-457     12-493
L$BMXZ  = ******   GX    12-497
L$BPAR  = ******   GX    14-644     14-645
L$BSA   = ******   GX    12-507     14-702
MBUF      000262 R      #7-187      17-961
MCFBF     000212 R      #7-162      11-416     11-439
MCFLG     000150 R      #7-154     *10-357    *10-365     14-634     15-783
MCLOA   = 000001        #7-153      10-357     10-365     14-634     15-783
MSGPRI    003152 R       7-134      7-137      10-394     10-399    #17-960
PCBA      000252 R      #7-172     *9-266      9-281     *9-285      9-288      10-348    *10-349     10-364    *10-387
                         10-388    *14-732     16-901
PR.LBL    001C02 R       9-273     #12-488
PR.PAR    001540 R       9-275      10-358    #14-632
PR.SUB    002334 R       14-695    #15-772
PR.XFR    003026 R       9-277      10-361    #16-891
PS.COM  = 000200         14-714     15-801
PS.NSF  = 000400         15-801
PS.SYS  = 000040         14-662     15-801
P.BLKS    000016         14-704     14-711    *15-793     15-829     15-833
P.BUSY    000024         14-716     14-720     14-723    *14-723     14-725    *14-727
P.MAIN    000012         14-718    *15-792
P.NAM     000004         10-334     10-336     14-651     14-653    *15-785    *15-786    *15-789    *15-790
P.REL     000014         14-672     14-699     15-810     15-814    *15-820    *15-825     15-832    *15-852     16-902
P.SIZE    000016         14-673
P.STAT    000030         14-662     14-714    *15-801
P.SUB     000010         15-811    *15-821    *15-853    *15-858
RMCLBL    000726 R       10-355    #11-454
R$$EIS  = ******         11-460     12-542     12-549     12-550     13-610     14-701
R$$MPL  = ******         8-215      14-656     14-667     14-675     15-795     15-837     16-891
R$$11D  = ******         12-542     13-610
SFIL      000624 R       10-351    #11-415
SYSFDB  = ******   GX   *9-292     *10-333    *10-383    *13-599    *14-650     14-680    *15-805    *16-901
TMPPCB    000152 R      #7-155     *10-348     10-373     10-387
TS$NHD  = ******   GX    12-491
T$FLAG    000044        #5-70
T$LIF     000013        #5-70
T$LIFL    000013        #5-70
T$LIFO    000013        #5-70
T$LIFS    000013        #5-70
T$LIN     000000        #5-70
T$LIPS    000006        #5-70
T$LLD     000012        #5-70
```

```
                                                                  G  5
VPIO       CREATED BY  MACRO  ON 28-AUG-85 AT 20:04     PAGE 3
```

```
                                                              F  6
VPRMPT     CREATED BY  MACRO  ON 29-JUN-85 AT 02:55     PAGE 2
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME      REFERENCES

DBGTP$          #6-79
ISTAT$          #5-65       6-79
MTRAN$          #6-79
PRMPT$          #5-65       6-83      #6-85     #6-86     #6-87     #6-88     #6-89     #6-90     #6-91     #6-92
                #6-93      #6-94      #6-95     #6-96     #6-97     #6-98     #6-99     #6-100    #6-101    #6-102
                #6-103     #6-104     #6-105    #6-106    #6-107    #6-108    #6-109    #6-110    #6-111    #6-112
                #6-113     #6-114     #6-115    #6-116    #6-117    #6-118    #6-119    #6-120    #6-121    #6-122
                #6-123     #6-124     #6-125    #6-126    #6-127    #6-128    #6-129    #6-130    #6-131    #6-132
                #6-133     #6-134     #6-135    #6-136    #6-137    #6-138    #6-139    #6-140    #6-141    #6-142
                #6-143     #6-144     #6-145    #6-146    #6-147    #6-148    #6-149    #6-150    #6-151    #6-152
                #6-153
```

```

000650  012700  000024'           30$:  MOV     #DSPTAB,R0        ; NOW TAKE THE ASCII NAME
000654  010701                          MOV     PC,R1             ; AND CONVERT TO RAD50
000656                                  CALL    $CAT5             ; STORING THE RAD50 NAME IN THE GLOBAL
000662  010167  000036'                 MOV     R1,PSYM           ; SYMBOL LOOKUP CONTROL BLOCK
000666  010701                          MOV     PC,R1             ;       ...
000670                                  CALL    $CAT5             ;       ...
000674  010167  000040'                 MOV     R1,PSYM+2         ;       ...
000700  116767  000000G 000025'         MOVB    $$DEV,DSPTAB+1    ; FORM DCB DISPATCH TABLE ADDRESS NAME
000706  116767  000001G 000026'         MOVB    $$DEV+1,DSPTAB+2 ;       ...
000714  012700  000024'                 MOV     #DSPTAB,R0        ; CONVERT TO RAD50
000720  010701                          MOV     PC,R1             ;       ...
000722                                  CALL    $CAT5             ;       ...
000726  010167  000120'                 MOV     R1,DTSYM          ; DEFAULT SECOND HALF OF NAME TO ''TBL''
000732  005000                          CLR     R0                ; SETUP ASCIZ FILE SPEC
000734  005001                          CLR     R1                ; FOR XXX.STB
000736  012702  000020'                 MOV     #STBEXT,R2        ;       ...
000742                                  CALL    $AZFS             ;       ...
000746  012700  000032'                 MOV     #PBLK,R0          ; GET LOOKUP CONTROL BLOCK ADDRESS
000752                                  CALL    $SYMBL            ; LOOK FOR GLOBAL SYMBOL
000756  103436                          BCS     STBERR            ; IF CS, FILE ERROR
                                                                  ; NOTE - R0 IS NOW 0
000760  016767  000124' 000000G         MOV     DTVAL,$$TBL       ; COPY DCB DISPATCH TABLE ADDRESS
000766  005767  000034'                 TST     PBLK+2            ; IS PROCESS DISPATCH TABLE DEFINED?
000772  001422                          BEQ     101$              ; NO, THEN FATAL ERROR

000774  012701  000164'                 MOV     #XBLK1,R1         ; POINT AT PROCESS SYMBOLS
001000  012702  000010G                 MOV     #$NTLHB+.CXSYM,R2 ; POINT AT CEX SYMBOLS

001004  005767  000054'                 TST     VECVAL            ; PROCESS VECTORED ?
001010  001402                          BEQ     50$               ; IF NO - BRANCH
001012  062702  000006                  ADD     #<.CXVSM-.CXSYM>,R2 ; POINT AT VECTORED SYMBOL VALUES

001016  026122  000010            50$:  CMP     10(R1),(R2)+      ; ALL SYMBOL DEFS. MUST
001022  001403                          BEQ     60$               ;  MATCH
001024  005761  000002                  TST     2(R1)             ;   UNLESS THEY ARE UNDEFINED
001030  001006                          BNE     111$              ; SYMBOL MISMATCH
001032  011101                    60$:  MOV     (R1),R1           ; LINK DOWN
001034  001370                          BNE     50$               ;  AND GO TEST NEW SYMBOL VALUE
001036                                  RETURN

                                  ;
                                  ; ERROR CONDITIONS
                                  ;
001040                            101$: EMSG$R  49                ; DISPATCH TABLE SYMBOL UNDEFINED
001046                            111$: EMSG$R  50                ; PROCESS/CEX INCOMPATIBLE
001054  016000  000220'           STBERR: MOV   ETAB1-2(R0),R0    ; GET ERROR MESSAGE CONTROL BLOCK ADDRESS
001060                                  CALLR   @(R0)+            ; PRINT ERROR MESSAGE
```

```
                                        .SBTTL  $ALECL - ALLOCATE ECL NODE COUNTERS

                                ;+
                                ;
                                ; $ALECL - ALLOCATE ECL NODE COUNTERS FROM POOL..
                                ;
                                ; INPUTS:
                                ;       $PDVA - ECL'S PDV ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       N$ENC    (ECL'S DDB) - APR6 VIRTUAL ADDRESS OF START OF NODE COUNTERS
                                ;       N$ENC+2  (ECL'S DDB) - APR6 VIRTUAL ADDRESS OF END OF NODE COUNTERS
                                ;       N$ENC+4  (ECL'S DDB) - APR BIAS FOR NODE COUNTERS
                                ;
                                ;-
004056                          $ALECL::CALL    $SAVAL                  ; SAVE ALL REGISTERS
004062                                  GETRV   $DECPT,#D$END           ; READ DECNET HOME BLOCK
004102  005000                          CLR     R0                      ; INIT REG
004104  156700  000041G                 BISB    $BFR+D$MNOD,R0          ; COPY NUMBER OF NODE COUNTERS
004110  010067  000000G                 MOV     R0,$$NODC               ; SAVE NUMBER OF NODE COUNTERS

004114  012701  000050                  MOV     #E$NLEN,R1              ; GET LENGTH OF ECL COUNTERS
004120                                  CALL    $MUL                    ; CALCULATE NUMBER OF BYTES
004124  062701  000077                  ADD     #77,R1                  ; ROUND TO BLOCK BYTE BOUNDARY
004130  042701  000077                  BIC     #77,R1                  ; ...
004134                                  ASR$    6,R1                    ; CONVERT TO BLOCK COUNT
004150  001522                          BEQ     40$                     ; BR IF NONE TO ALLOCATE
004152                                  CALL    $ALN18                  ; ALLOCATE COUNTERS FROM NON-UMR SPACE
004156  103520                          BCS     101$                    ; BR IF ERROR
004160  016705  000000G                 MOV     $PDVA,R5                ; GET ECL'S PDV ADDRESS
004164  016505  000000G                 MOV     Z.DAT(R5),R5            ; GET ECL'S DDB ADDRESS
004170                                  GETRV   R5,#N$ELEN              ; READ IN ECL'S DDB
004206  010067  000046G                 MOV     R0,N$ENC+4+$BFR         ; STORE APR BIAS NODE COUNTERS

004212  005001                          CLR     R1                      ; POINT TO START OF COUNTERS
004214  052701  120000                  BIS     #120000,R1              ; SET CORRECT APR BIAS
004220  005004                          CLR     R4                      ; POINT TO START OF COUNTERS
004222                                  BIAS    R4                      ; ... (ECL NEEDS APR6 BIAS)
004232  010467  000042G                 MOV     R4,N$ENC+$BFR           ; STORE START OF COUNTERS
004236                                  PUTRC                           ; REWRITE UPDATED DDB

004242  016746  000000G                 MOV     $$NODC,-(SP)            ; GET NUMBER OF NODE COUNTERS
004246  010102                          MOV     R1,R2                   ; POINT TO START OF COUNTERS
004250                          20$:    GETAD   R2,#E$NLEN,R0           ; READ IN NEXT COUNTER BLOCK
004272  062702  000050                  ADD     #E$NLEN,R2              ; POINT TO NEXT COUNTER BLOCK IN LIST
004276  012701  000000G                 MOV     #$BFR,R1                ; POINT TO COUNTER BLOCK
004302  010204                          MOV     R2,R4                   ; COPY ADDRESS OF NEXT BLOCK
004304                                  BIAS    R4                      ; SET CORRECT APR BIAS FOR ECL
004314  010421                          MOV     R4,(R1)+                ; STORE LINK WORD
004316  012703  000023                  MOV     #<<E$NLEN-2>/2>,R3      ; GET NUMBER OF WORDS TO ZERO
004322  005021                  30$:    CLR     (R1)+                   ; ZERO THE BLOCK
004324                                  SOB     R3,30$                  ; ...
004330                                  PUTAD                           ; REWRITE COUNTER BLOCK
004334  005316                          DEC     (SP)                    ; MORE TO ALLOCATE?
004336  003344                          BGT     20$                     ; BR IF YES
004340  005726                          TST     (SP)+                   ; CLEAN UP STACK
004342  162701  000050                  SUB     #E$NLEN,R1              ; POINT TO START OF PREVIOUS BLOCK
```

```
                                                                                  F  9
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05      PAGE 7
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE             REFERENCES

$PUTRC  = ******  GX        12-660     13-725     13-791     14-837     14-847     15-930     15-949     16-1034    17-1119
                            17-1143
$RADDR  = ******  GX       *7-270     *7-308     *7-311     *8-351     *11-571    *11-577    *11-590    *12-618    *12-640
                           *12-681    *13-707    *13-720    *13-750    *13-774    *13-788    *14-817    *14-835    *14-838
                           *14-845    *15-877    *15-899    *15-917    *15-928    *15-942    *15-951    *16-989    *16-1010
                           *16-1053   *16-1065   *17-1096   *17-1111   *17-1123   *17-1141
$RBLCK  = ******  GX       *7-308     *7-311     *11-576    *12-681    *13-750    *13-774    *14-838    *15-899    *15-917
                           *15-951    *16-1053   *17-1123
$ROL      001452 RG        #12-618
$SAVAL  = ******  GX        14-812     16-987     17-1095
$SLTA   = ******  GX        11-530
$SLTMA  = ******  GX        11-535
$SQSRV    000136 R         #6-179      9-392      9-393
$SYMBL  = ******  GX        10-479
$T1SCL  = ******  GX       *9-386     *9-397
$T100C  = ******  GX       *9-385     *9-396
$XFER     000000 RG        #7-245
$$DEV   = ******  GX        10-468     10-469
$$HTIM  = ******  GX        11-586
$$LTIM  = ******  GX        11-587
$$NBRA  = ******  GX        11-588
$$NODC  = ******  GX       *17-1099    17-1121
$$RPRI  = ******  GX        11-589
$$TBL   = ******  GX       *10-482
.CXALL    000062           #5-76
.CXCSR    000004           #5-76
.CXLBR    000060           #5-76
.CXLB1    000030           #5-76
.CXLB2    000044           #5-76
.CXLEN    000064           #5-76
.CXPCB    000006           #5-76
.CXSYM    000010           #5-76       10-487     10-491
.CXUNL    000024           #5-76
.CXVEC    000026           #5-76       8-340     *8-341
.CXVSM    000016           #5-76       10-491
.MGMGE  = ******  GX        7-279      9-379
.NTGCD    000000           #5-76
.NTPCB    000000           #5-76
.NTUMR    000002           #5-76
.PLGTH  = ******  GX        7-270
.SBPCB    000066           #5-76
```

```
                                                                                  G  9
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05      PAGE 8
```

```
                                        .SBTTL  SETUP DEFAULT FILE NAME
                                ;+
                                ; $DFN - SETUP DEFAULT FILE NAME
                                ;
                                ; INPUTS:
                                ;       R0=RAD50 FILE NAME, FIRST HALF
                                ;       R1=RAD50 FILE NAME, SECOND HALF ($DFN2 ONLY)
                                ;
                                ; OUTPUTS:
                                ;       R0,R1,R2=DESTROYED
                                ;-
001160                          $DFN::
001160  005001                          CLR     R1              ; NO SECOND HALF
001162                          $DFN2::
001162  010146                          MOV     R1,-(SP)        ; SAVE SECOND HALF
001164  010001                          MOV     R0,R1           ; COPY FIRST HALF
001166  012700  001552'                 MOV     #$DFNAM,R0      ; GET DEFAULT FILE NAME ADDRESS
001172                                  CALL    10$             ; CONVERT TO ASCII
001176  012601                          MOV     (SP)+,R1        ; RESTORE SECOND HALF
001200  001407                          BEQ     20$             ; IF ZERO, USE FIRST HALF ONLY
001202                          10$:    CALL    $C5TA           ; CONVERT TO ASCII
001206  122740  000040          15$:    CMPB    #' ,-(R0)       ; SKIP TRAILING BLANKS
001212  001775                          BEQ     15$             ;       ...
001214  005200                          INC     R0              ;       ...
001216  105010                          CLRB    (R0)            ; MAKE IT AN ASCIZ STRING
001220                          20$:    RETURN
```

```
 004032  000000                          $PSIPT::.WORD   0                       ; POINTER TO PSI HOME BLOCK
 004034  000000                          $SNAPT::.WORD   0                       ; POINTER FOR SNA
 004036  000000                          $DDFNC::.WORD   0                       ; POWERFAIL FLAG
 004040  000000                          $CCBLH::.WORD   0                       ; POINTER TO FIRST FREE CCB
 004042  000000                          $SDBCT::.WORD   0                       ; COUNTER OF CURRENT SDB'S IN POOL
 004044  000000                          $SDBLH::.WORD   0                       ; POINTER TO FIRST FREE SDB
 004046  000000                                 .WORD   0                       ;    ...
 004050  000000                          $SDBAF::.WORD   0                       ; COUNTER OF # OF SDB ALLOCATION FAILURES
 004052  000000                          $RDBCT::.WORD   0                       ; COUNTER OF CURRENT RDB'S IN POOL
 004054  000000                          $RDBLH::.WORD   0                       ; POINTER TO FIRST FREE RDB
 004056  000000                                 .WORD   0                       ; SECOND HALF OF ADDRESS DOUBLEWORD
 004060  000000                          $RDBAF::.WORD   0                       ; COUNTER OF # OF RDB ALLOCATION FAILURES
 004062  000000                          $RDQSL::.WORD   0                       ; SYSTEM LINE NUMBER TO CHECK ON NEXT BUFFER RETURN
 004064  000000                          $RDQCT::.WORD   0                       ; NUMBER OF BUFFER WAIT REQUESTS AND FLAG
 004066  000000                          $NTLPT::.WORD   0                       ; POINTER TO NTL HOME BLOCK
 004070  000000                          $CXOPT::.WORD   0                       ; COMM/EXEC OPTIONS INCLUDED
 004072  000000                          $NBIAS::.WORD   0                       ; NON-UMR MAPPED POOL POINTER
 004074  000000                          $QBIAS::.WORD   0                       ; UMR-MAPPED POOL POINTER
 004076  000000                          $QSTRT::.WORD   0                       ; ORIGINAL BIAS OF UMR-MAPPED PORTION OF POOL

                                                .IF NDF R$$MPL                                                  ;[MP01]
 004100  000000                          $PUMR:: .WORD   0                       ; STARTING UMR BIAS                     ;[MP01]
                                                .ENDC                                                           ;[MP01]

 004102  000000                          $PWRF1::.WORD   0                       ; POWER FAIL FLAG
 004104  000000                          $XBIAS::.WORD   0                       ; BIAS OF EXTENDED SINGLE WORD POOL
 004106                                  $PAVL::
 004106  000001                          $XAVL:: .WORD   1                       ; POINTER TO 1ST FREE BLOCK IN EXT POOL
 004110  000000                          $NMLST::.WORD   0
 004112  000000                          $OBJHD::.WORD   0                       ; OBJECT TABLE LISTHEAD
 004114                                  $TK100::                                ; NUMBER OF CLOCK TICKS IN 100 MSEC
 004114  000000                          $TCK50::.WORD   0                       ; NUMBER OF CLOCK TICKS IN 50 MSEC
 004116     000                          $SQRCM::.BYTE   0                       ; SQUARE ROOT LIMIT FOR BUFF ALLOCATION
 004117     000                                 .BYTE   0                       ; NUMBER OF ACTIVE COMPONENTS
 004120  000000                          $MAXOV::.WORD   0                       ; MAXIMUM PROTOCOL OVERHEAD SIZE
 004122  000000                          $LGPDV::.WORD   0                       ; PDV INDEX OF EVENT LOGGER
 004124  000000                          $LGDDB::.WORD   0                       ; ADDRESS OF EVENT LOGGER DATABASE
 004126  000000                          $FILHD::.WORD   0                       ; EVENT FILTER LISTHEAD
 004130  000000                          $LGSTT::.WORD   0                       ; EVENT LOGGER STATUS

 004132  000000  000000  000000          $EVDSC::.WORD   0,0,0,0,0,0,0,0         ; EVENT DESCRIPTOR BLOCK

 004152  000000                          $LGFNB::.WORD   0                       ; EVENT LOGGER FILENAME BLOCK - FILE ID
 004154  000000                                 .WORD   0                       ;    ...
 004156  000000                                 .WORD   0                       ;    ...
 004160  021265  055254  057330                 .RAD50  /EVENTLOG/              ; FILENAME
 004166  075273                                 .RAD50  /SYS/                   ; FILE TYPE
 004170  000000                                 .WORD   0                       ; FILE VERSION NUMBER
 004172     114     102                         .ASCII  /LB/                    ; DEVICE NAME
 004174  000000                                 .WORD   0                       ; AND UNIT NUMBER
 004176  000000                          $LGUIC::.WORD   0                       ; UIC FOR LOGGING FILE
 004200  051646  131574                  $LGMON::.RAD50  /MON.../                ; MONITOR TASK NAME
 004204  000000                          $LGCON::.WORD   0                       ; CONSOLE DEVICE NAME
 004206  000000                                 .WORD   0                       ; AND UNIT
 004210  000000                          $CELFN: .WORD   0                       ; CEX LOGGING FUNCTION CODE
 004212  000000  004212'                 $NMCLH::.WORD   0,.-2                   ; NETWORK MANAGEMENT COMPLETION LISTHEAD

```

```
                                        .SBTTL  $SWOVL - SWITCH OVERLAYS
                                ;+
                                ;
                                ;       $SWOVL - SWITCH OVERLAYS
                                ;
                                ;       THIS ALLOWS A OVERLAY SEGMENT AT THE SAME LEVEL TO BE CALLED
                                ;
                                ;
                                ; INPUTS:
                                ;       R1 - NUMBER IDENTIFYING WHICH ROUTINE TO CALL
                                ;           = 1 -- CALL $FPDV
                                ;           = 2 -- CALL $VACP
                                ;
                                ; OUTPUT:
                                ;       CALLS THE SPECIFIED ROUTINE,
                                ;       AND RETURNS TO CALLER
                                ;
                                ;-

005716  005301                  $SWOVL::DEC     R1              ; CHECK IF R1 WAS 1
005720  001012                          BNE     10$             ; NO CHECK IF R1 WAS 2
005722                                  CALL    $FPDV           ; CALL $FPDV
005726  103004                          BCC     5$              ; BR IF FOUND IT
005730  005767  176076                  TST     $PSIPT          ; IS PSI LOADED?
005734  001001                          BNE     5$              ; BR IF YES - NOT DLX ONLY SYSTEM (CARRY CLEAR)
005736  000261                          SEC                     ; ELSE INDICATE DLX ONLY SYSTEM
005740  006101                  5$:     ROL     R1              ; CARRY SIGNIFICANT -- SAVE IT
005742  000167  000000G                 JMP     $RET1           ; AND RETURN
005746                          10$:    CALL    $VACP           ; MUST HAVE BEEN = 2 -- CALL $VACP
005752  000167  000000G                 JMP     $RET2
```

```
                                                                F 13
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 8
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE             REFERENCES

S.FNB   = 000036           6-102      6-114
S.FTYP  = 000002           6-102      6-102      6-114      6-114
S.LEN     000004  G       #5-78
S.MPR     003550 R        #16-658
S.NMST    000002  G       #5-78
S.OWNR    000003  G       #5-78
TIOUT     000000 R        #6-96      *14-418    *14-419     14-420
TOPFLG    000754 RG       #7-179
TY$CHA  = 000400          #5-80
TY$EVE  = 001000          #5-80
TY$STA  = 002000          #5-80
TY$SUM  = 000000          #5-80
UMRFLG    000776 RG       #7-188
UMRPTR    000774 RG       #7-187
VF.ACP  = 001000  G       #15-604
VF.ADD  = 000400  G       #15-603
VF.DLX  = 004000  G       #15-606
VF.EAD  = 020000  G       #15-608
VF.ECH  = 000020  G       #15-599
VF.FIX  = 000100  G       #15-601
VF.INI  = 000040  G       #15-600
VF.NTI  = 000001  G       #15-595
VF.ON   = 000002  G       #15-596
VF.PEM  = 002000  G       #15-605
VF.PSI  = 040000  G       #15-609
VF.SED  = 100000  G       #15-610
VF.SKA  = 010000  G       #15-607
VF.SS   = 000010  G       #15-598
VF.TRA  = 000004  G       #15-597
VF.VDT  = 000200  G       #15-602
V1.NSY  = 000001  G       #15-621
WC.CNT  = 000400          #5-80
WC.EVE  = 004000          #5-80
WC.TRI  = 002000          #5-80
WC.UNT  = 001000          #5-80
W.TSEF  = 000002            6-102
X$$MCB  = ******            5-79      5-79
ZF.COU  = 001000  G       #5-79
ZF.DDM  = 000001  G       #5-79
ZF.DIA  = 004000  G       #5-79
ZF.DLC  = 000002  G       #5-79
ZF.DVP  = 100000  G       #5-79
ZF.INI  = 040000  G       #5-79
ZF.KMX  = 000020  G       #5-79
ZF.LLC  = 000004  G       #5-79
ZF.LMC  = 000100  G       #5-79
ZF.MAN  = 020000  G       #5-79
ZF.MFL  = 000010  G       #5-79
ZF.MTM  = 000400  G       #5-79
ZF.MUX  = 000040  G       #5-79
ZF.PSE  = 002000  G       #5-79
ZF.SLI  = 010000  G       #5-79
```

G 13

```
                                              .SBTTL  FUNCTION TABLES

                                      ;
                                      ; LOCAL DATA
                                      ;
 000000                                       .PSECT  DATA,D
                                              .NLIST  BEX
                                      ;
                                      ; LOOKUP TABLE USING FUNCTION CODE AS INDEX
                                      ;
 000000                               FTAB:
 000000  000076'                              .WORD   .SCEX           ; 1 - NODE ON (NTINIT)
 000002  000104'                              .WORD   .SPRO           ; 2 - SET PROCESS
 000004  000116'                              .WORD   .SLIN           ; 3 - SET LINE
 000006  000132'                              .WORD   .CCEX           ; 4 - NODE OFF
 000010  000136'                              .WORD   .CPRO           ; 5 - CLEAR PROCESS
 000012  000146'                              .WORD   .CLIN           ; 6 - CLEAR LINE
 000014  000000                               .WORD   0               ; 7 - RELOAD KMC MICROCODE POWERFAIL
 000016  000156'                              .WORD   .NOP            ; 10- TEST ROUTINES
 000020  000156'                              .WORD   .NOP            ; 11- EXIT VNP
 000022  000156'                              .WORD   .NOP            ; 12- HELP
 000024  000230'                              .WORD   .RNTI           ; 13- SET SYSTEM
 000026  000162'                              .WORD   .SNOD           ; 14- SET NODE
 000030  000162'                              .WORD   .CNOD           ; 15- CLEAR NODE
 000032  000224'                              .WORD   .SALI           ; 16- SET ALIAS
 000034  000224'                              .WORD   .CALI           ; 17- CLEAR ALIAS
 000036  000170'                              .WORD   .SLOG           ; 20- SET LOGGING
 000040  000170'                              .WORD   .CLOG           ; 21- CLEAR LOGGING
 000042  000224'                              .WORD   .SOBJ           ; 22- SET OBJECT
 000044  000224'                              .WORD   .COBJ           ; 23- CLEAR OBJECT
 000046  000224'                              .WORD   .DCEX           ; 24- SHOW SYSTEM
 000050  000224'                              .WORD   .DLIN           ; 25- SHOW LINE
 000052  000224'                              .WORD   .DPRO           ; 26- SHOW PROCESS
 000054  000224'                              .WORD   .DNOD           ; 27- SHOW NODE
 000056  000224'                              .WORD   .DALI           ; 30- SHOW ALIAS
 000060  000224'                              .WORD   .DLOG           ; 31- SHOW LOGGING
 000062  000224'                              .WORD   .DOBJ           ; 32- SHOW OBJECT
 000064  000204'                              .WORD   .SCIR           ; 33- SET CIRCUIT
 000066  000000                               .WORD   0               ; 34- CLEAR CIRCUIT
 000070  000224'                              .WORD   .DCIR           ; 35- SHOW CIRCUIT
 000072  000216'                              .WORD   .SMOD           ; 36- SET MODULE
 000074  000176'                              .WORD   .DMOD           ; 37- SHOW MODULE
         000037                       FMAX=.-FTAB/2


                                      ;
                                      ; SUBROUTINE TABLES
                                      ;
 000076  001532'                      .SCEX:  .WORD   CLRPP           ; CLEAR PROCESS PARAMETERS
 000100  000124'  000000                      .WORD   SCEX,0
 000104  000176'                      .SPRO:  .WORD   CEXLD           ; MAKE SURE COMM EXEC IS LOADED
 000106  000474'                              .WORD   PDVA            ; FIND PDV ADDRESS
 000110  000552'                              .WORD   CHKPP           ; CHECK PROCESS PARAMETERS
 000112  000670'  000000                      .WORD   SPRO,0
 000116  000176'                      .SLIN:  .WORD   CEXLD           ; MAKE SURE COMM EXEC IS LOADED
 000120  000774'                              .WORD   SLTA            ; FIND SLT ADDRESS
 000122  001264'                              .WORD   CHKLP           ; CHECK LINE PARAMETERS
 000124  001532'                              .WORD   CLRPP           ; CLEAR PROCESS PARAMETERS
```

```
    523                                              .SBTTL  SPRO - CHECK SET PROCESS PARMS
    524                                      ;+
    525                                      ; SPRO - CHECK "SET PROCESS" REQUEST BLOCK
    526                                      ;
    527                                      ; INPUTS:
    528                                              $PDVA=PDV ADDRESS
    529                                      ;
    530                                      ; OUTPUTS:
    531                                              R4 CLEARED IF NO WORK TO DO
    532                                      ;-
    533
    534 000670  012700  000356'              SPRO:   MOV     #PBITS,R0               ; DO BIT TWIDDLING
    535 000674                                       CALL    $BITS                   ;
    536 000700  032712  000000G                      BIT     #FL.LLC,(R2)            ; IS THIS AN LLC ?
    537 000704  001403                               BEQ     5$                      ; NO
    538 000706  052767  000000G 000000G              BIS     #MS.LLC,$MISS           ; INDICATE LLC INFO MISSING
    539 000714  016700  000000G              5$:     MOV     $PDVA,R0                ; GET PDV ADDRESS
    540
    541 000720  005760  000000G                      TST     Z.PCB(R0)               ; IS PROCESS CORE-RESIDENT? (CLEARS C-BIT)
    542 000724  001407                               BEQ     20$                     ; IF EQ, NO
    543 000726  032712  000000G                      BIT     #FL.LLC,(R2)            ; IS THIS AN LLC?
    544 000732  001403                               BEQ     10$                     ; IF EQ, NO
    545 000734  005760  000000G                      TST     Z.DAT(R0)               ; IS LINE TABLE PRESENT? (CLEARS C-BIT)
    546 000740  001401                               BEQ     20$                     ; IF EQ, NO
    547 000742  005004                       10$:    CLR     R4                      ; NO WORK TO DO!
    548 000744                               20$:    RETURN
    936                                                  .SBTTL  CCEX - CHECK CLEAR SYS PARMS
    937                                          ;+
    938                                          ; CCEX - CHECK "CLEAR CEX" REQUEST BLOCK
    939                                          ;
    940                                          ; INPUTS:
    941                                          ;        R4 = FUNCTION CODE
    942                                          ;
    943                                          ; OUTPUTS:
    944                                          ;        NONE
    945                                          ;-
    946
    947 002536                                   CCEX:   CALL    $SAVAL          ; SAVE REG'S
    948 002542  032767  000000G 000000G                  BIT     #FE.CEX,$FMASK  ; IS THE COMM EXEC LOADED ?
    949 002550  001002                                   BNE     10$             ; IF NE, YES .. OKAY
    950 002552  005004                                   CLR     R4              ; ELSE, NO WORK TO DO !
    951 002554  000433                                   BR      30$             ; AND RETURN
    952
    953 002556  012700  002646'                  10$:    MOV     #NTINIT,R0      ; POINT AT RAD50 NAME
    954 002562                                           CALL    TSKSCH          ; GET TCB ADDRESS
    955 002566  103425                                   BCS     20$             ; NTINIT NOT EVEN INSTALLED
    956 002570  012700  114224                           MOV     #^RXPT,R0       ; IS THIS DLX ONLY SYSTEM ?
    957 002574                                           CALL    $FPDV           ; LOOK FOR PDV
    958 002600  103020                                   BCC     20$             ; NOT DLX ONLY, FOUND XPT PDV -- CONTINUE
    959
    960 002602  016700  000012G                  15$:    MOV     T.RCVL+$BFR,R0  ; GET PACKET ADDRESS
    961
    962                                                  .IF DF  R$$MPL                                                  ;[MP01]
    963                                                  BIT     #DF.POL,$DFLAG  ; SECONDARY POOL SUPPORTED              ;[MP01]
    964                                                  BEQ     16$             ; IF NO - BRANCH                        ;[MP01]
    965                                                  GETAD   #140000,#DSIZE,R0 ; READ PACKET FROM SECONDARY POOL     ;[MP01]
    966                                                  BR      18$             ; CONTINUE                              ;[MP01]
    967
    968                                                  .IFTF                                                           ;[MP01]
    969 002606                                   16$:    GETRV   R0,#DSIZE       ; READ SEND DATA PACKET
    970                                                  .ENDC                                                           ;[MP01]
    971
    972 002624  032767  010000  000006G          18$:    BIT     #LS.CXO,LR.STS+6+$BFR ; IS EXEC STATE ON ?
    973 002632  001403                                   BEQ     20$             ; IS NO - BRANCH
    974
    975 002634                                           EMSG$R  Z8              ; ELSE, NODE ON ERROR
    976 002642  000241                           20$:    CLC                     ; ALL CLEAR
    977 002644                                   30$:
    978 002644                                           RETURN
    979
    980 002646  055251  054374                   NTINIT: .RAD50  /NTINIT/        ; RAD50 NETWORK INITIALIZATION TASK NAME
    981
    982                                          ;+
    983                                          ;        TSKSCH  - SEARCH TCB'S FOR SPECIFIED TASK
    984                                          ;
    985                                          ; INPUTS:
    986                                          ;        R0      - POINTER TO RAD50 TASK NAME
    987                                          ;
    988                                          ; OUTPUTS:
    989                                          ;        $RADDR  - ADDRESS OF TCB
    990                                          ;        $BFR    - IMAGE OF TCB
    991                                          ;        C-BIT SUCCESS/FAILURE
    992                                          ;-
```

```
                                                                  G  1
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 3
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

ERR17       001306 R        #10-643       10-643        10-643
ERR2        000370 R        #10-630       10-630        10-630
ERR20       001336 R        #10-644       10-644        10-644
ERR21       001372 R        #10-645       10-645        10-645
ERR22       001450 R        #10-646       10-646        10-646
ERR23       001510 R        #10-647       10-647        10-647
ERR24       001566 R        #10-648       10-648        10-648
ERR25       001622 R        #10-649       10-649        10-649
ERR26       001654 R        #10-650       10-650        10-650
ERR27       001676 R        #10-651       10-651        10-651
ERR3        000452 R        #10-631       10-631        10-631
ERR30       001732 R        #10-652       10-652        10-652
ERR31       001744 R        #10-653       10-653        10-653
ERR32       002000 R        #10-654       10-654        10-654
ERR33       002042 R        #10-655       10-655        10-655
ERR34       002102 R        #10-656       10-656        10-656
ERR35       00214[illegible] R        #10-657       10-657        10-657
ERR36       002200 R        #10-658       10-658        10-658
ERR37       002232 R        #10-659       10-659        10-659
ERR4        000522 R        #10-632       10-632        10-632
ERR40       002264 R        #10-660       10-660        10-660
ERR41       002320 R        #10-661       10-661        10-661
ERR42       002364 R        #10-662       10-662        10-662
ERR43       002420 R        #10-663       10-663        10-663
ERR44       002454 R        #10-664       10-664        10-664
ERR45       002512 R        #10-665       10-665        10-665
ERR5        000566 R        #10-633       10-633        10-633
ERR6        000622 R        #10-634       10-634        10-634
ERR7        000674 R        #10-635       10-635        10-635
EVENT       000046 R        #6-125        77-3444       78-3465       79-3499       79-3500       79-3501       79-3502
EXT         014202 RG       #90-3842
EXTN     =  000003          #7-195        90-3842
E$NBR       000014          #5-69
E$NBS       000020          #5-69
E$NCR       000034          #5-69
E$NCS       000036          #5-69
E$NIC       000044          #5-69
E$NLEN      000050          #5-69         24-1442
E$NLLA      000012          #5-69
E$NLNK      000000          #5-69
E$NML       000040          #5-69
E$NMR       000024          #5-69
E$NMS       000030          #5-69
E$NNOD      000002          #5-69
E$NRT       000042          #5-69
E$NRTP      000005          #5-69
E$NSEG      000010          #5-69
E$NTIM      000046          #5-69
E$NUSE      000004          #5-69
E$STRT      000006          #5-69
E.CON       012676 R        #79-3487
E.FIL       012706 R        #79-3489
```

```
                                                                  H  1
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 4
```

```
                                                                  G  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85  AT 02:50     PAGE 16

SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE            REFERENCES

                         *104-4653   104-4655    104-4657  *104-4665  *104-4669  *104-4674
UISAR0   = 177640        #5-75
UISAR1   = 177642        #5-75
UISZ     = 000014  G     #7-175      103-4548
UMRBF      000006 R      #6-117      11-696
UMRFLG   = ******  GX     8-409      11-688     *24-1457
UMRPTR   = ******  GX     8-375     *11-685     *11-695     16-1015
UNT        000061 R      #6-128
USRUIC     017142 RG     #104-4647
USRUI1     017134 RG     #104-4646
VF.DLX   = ******  GX     8-523      8-527
VF.NTI   = ******  GX     110-4878   110-4881
VF.SS    = ******  GX     17-1083
WC.CNT   = 000400        #5-78
WC.EVE   = 004000        #5-78
WC.TRI   = 002000        #5-78
WC.UNT   = 001000        #5-78
XPCIR      004600 R      #25-1499
XPTLLN     000066 R      #6-134      24-1385     24-1415   *25-1499
X$$MCB   = ******         5-72       5-72
X.FLG      010244 R      #66-2988
X.FLG1     010336 R      #66-3006
ZF.COU   = 001000        #5-72
ZF.DDM   = 000001        #5-72
ZF.DIA   = 004000        #5-72
ZF.DLC   = 000002        #5-72
ZF.DVP   = 100000        #5-72
ZF.INI   = 040000        #5-72
ZF.KMX   = 000020        #5-72
ZF.LLC   = 000004        #5-72
ZF.LMC   = 000100        #5-72
ZF.MAN   = 020000        #5-72
ZF.MFL   = 000010        #5-72       28-1688     56-2647    106-4736
ZF.MTM   = 000400        #5-72
ZF.MUX   = 000040        #5-72
ZF.PSE   = 002000        #5-72       106-4751
ZF.SLI   = 010000        #5-72
ZF.TIM   = 000200        #5-72
ZF.X3P   = 000000        #5-72
ZS.ASN   = 100000        #5-72
ZS.BSY   = 140000        #5-72
Z.AVL      000014        #5-72
Z.DAT      000016        #5-72
Z.DSP      000000        #5-72       5-72
Z.FLG      000010        #5-72      *28-1688    *56-2647    106-4736   106-4751
Z.LEN    = 000016        #5-72
Z.LLN      000006        #5-72
Z.MAP      000020        #5-72
Z.NAM      000004        #5-72       56-2642     59-2695    96-4183    107-4788   108-4826   109-4857
Z.PCB      000012        #5-72       106-4734
Z.SCH      000007        #5-72
$ALL15   = ******  GX     11-693     26-1535
```

```
                                                                  H  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85  AT 02:50     PAGE 17
```

```
                                .SBTTL  MACRO DEFINITIONS
                        ;***
                        ; LIBRARY MACROS
                        ;***
                                .MCALL  HWDDF$,PC8DF$,GETRV,PUTRC,SWTCHI,ASL$,ASR$,GETAD,PUTAD,DHBDF$
                                .MCALL  CTRDF$,CACDF$,ECDD8$,ERMSG$,ERBLK$,ERRPT$

                                .ENABL  LC

                        ;***
                        ; LIBRARY SYMBOLS
                        ;***

 000000                         HWDDF$                          ; FEATURES MASK 8IT DEFINITIONS
 000000                         PCBDF$                          ; PARTITION CONTROL BLOCK OFFSETS
 000000                         DHBDF$                          ; DECNET HOME BLOCK OFFSETS
 000000                         CTRDF$                          ; DEFINE COUNTER BLOCK OFFSETS
 000000                         CACDF$                          ; DEFINE CACHE OFFSETS
 000000                         ECDDB$                          ; DEFINE ECL DATA BASE OFFSETS


                        ;***
                        ; LOCAL MACROS
                        ;***

                        ;
                        ; SHIFT LEFT
                        ;
                                .MACRO  ASL$    COUNT,REG
                                .IF DF  R$$EIS
                                ASH     #'COUNT',REG
                                .IFF
                                .REPT   COUNT
                                ASL     REG
                                .ENDR
                                .ENDC
                                .ENDM   ASL$
```

```
    580                                                 .SBTTL  EXTXPT - EXTEND XPT'S DDB
    581                                         ;+
    582                                         ; EXTXPT - EXTEND XPT'S DATA BASE FOR ROL & ROA
    583                                         ;
    584                                         ; INPUTS:
    585                                         ;       R2 = CURRENT EXTEND SIZE
    586                                         ;       $LDSIZ = PROCESS LOAD SIZE
    587                                         ;       $NAME = RAD50 PROCESS NAME
    588                                         ;
    589                                         ; OUTPUTS:
    590                                         ;       R2 = NEW EXTEND SIZE
    591                                         ;       $LDSIZE = NEW LOAD SIZE
    592                                         ;
    593                                         ;-
    594
    595 001424                                  EXTXPT:
    596 001424  010146                                  MOV     R1,-(SP)                ; SAVE BLOCK SIZE
    597 001426  022767  114224  000000G                 CMP     #^RXPT,$NAME            ; IS THIS PROCESS XPT ?
    598 001434  001037                                  BNE     20$                     ; IF NE, NO
    599 001436                                          GETRV   $DECPT,#D$END           ; READ DECNET HOME BLOCK
    600 001456  005767  000060G                         TST     $BFR+D$NN               ; ENDNODE ?
    601 001462  001024                                  BNE     20$                     ; IF NOT - BRANCH
    602
    603 001464  005000                                  CLR     R0                      ; INIT REG
    604 001466  156700  000040G                         BISB    $BFR+D$MLL,R0           ; COPY MAX LOGICAL LINKS
    605 001472  012701  000012                          MOV     #C$LEN,R1               ; GET CACHE ENTRY SIZE
    606 001476                                          CALL    $MUL                    ; CALCULATE TOTAL SIZE
    607
    608 001502  062701  000077                          ADD     #77,R1                  ; ROUND UP TO NEXT BLOCK
    609 001506  042701  000077                          BIC     #77,R1                  ; ...
    610 001512                                          ASR$    6,R1                    ; ELSE, CONVERT BYTES TO BLOCKS
    611 001526  060102                                  ADD     R1,R2                   ; INCREASE EXTEND SIZE
    612 001530  060167  000260'                         ADD     R1,$LDSIZ               ; AND LOAD SIZE
    613 001534  012601                          20$:    MOV     (SP)+,R1                ; RESTORE BLOCK SIZE
    614 001536                                          RETURN
```

```
                                                                                G  5
VPIO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:04       PAGE 3
SYMBOL  CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

T$LLDC     000045           #5-70
T$LLDL     000012           #5-70
T$LLDO     000012           #5-70
T$LLDS     000012           #5-70
T$LLEN     000046           #5-70
T$LOPR     000002           #5-70
T$LTCL     000024           #5-70
T$LTIM     000026           #5-70
T$LTPR     000014           #5-70
T$LTPS     000020           #5-70
T$NAPL     000004           #5-70
T$NFE      000000           #5-70
T$NLEN     000010           #5-70
T$NNUL     000002           #5-70
T$NOPL     000006           #5-70
T$NRNI     000042           #5-70
T$NRPL     000005           #5-70
T$NRUL     000007           #5-70
T$NVR      000001           #5-70
T$RPRI     000040           #5-70
T$SVC      000034           #5-70
T$T5       000030           #5-70
T$T6       000032           #5-70
UNAMC      000154 R         #7-156      11-427
UNAOP      000050 R         #7-134      10-394
UNARE      000136 R         #7-137      10-399
VF.SS    = ******    GX       10-325    14-676     15-834
Z.NAM    = ******    GX       10-341    17-937
$ALL15   = ******    GX       15-773
$BFR     = ******    GX      *9-294    *10-328     10-329     10-334     10-336    *10-385     13-600     13-604    *14-646
                              14-647    14-651     14-653     14-662     14-672     14-673     14-699     14-704     14-711
                              14-714    14-716     14-718     14-720     14-723    *14-723     14-725    *14-727     15-776
                             *15-785   *15-786    *15-789    *15-790    *15-792    *15-793    *15-801     15-810     15-811
                              15-814   *15-820    *15-821    *15-825     15-829     15-832     15-833    *15-852    *15-853
                             *15-858    16-902
$CLDEQ   = ******    GX        9-278    10-363     10-396
$CLIOS   = ******    GX        9-269    10-397     12-567     16-911
$CLOPE   = ******    GX        9-268    10-352
$DDMPV   = ******    GX       10-340
$DEA16   = ******    GX       15-870
$DECPT   = ******    GX       13-599
$DFUIC   = ******    GX       11-417
$ERMSG   = ******    GX       17-962
$FLAGS   = ******    GX       10-343    12-505     12-520    *14-632     14-664     14-668    *14-693     14-697     15-823
                              15-847
$FMASK   = ******    GX        9-290    10-381     14-707
$GETRV   = ******    GX        9-292    10-333     10-383     13-599     14-650     14-719     14-722     14-726     15-809
                              15-813    15-819     15-828     15-831     15-851     15-857     16-901
$ICBLN   = ******    GX       15-839
$IOSB    = ******    GX       10-397    12-567     16-911
$LDSIZ     000260 RG        #7-182       9-297     *9-299    *11-458    *12-495    *13-612
$LLEN    = ******    GX      *11-461   *12-551
```

```
**FILE**ID**VPRO




VV      VV   PPPPPPPP     RRRRRRRR       000000
VV      VV   PPPPPPPP     RRRRRRRR       000000
VV      VV   PP      PP   RR      RR   00      00
VV      VV   PP      PP   RR      RR   00      00
VV      VV   PP      PP   RR      RR   00      00
VV      VV   PP      PP   RR      RR   00      00
VV      VV   PPPPPPPP     RRRRRRRR     00      00
VV      VV   PPPPPPPP     RRRRRRRR     00      00
VV      VV   PP           RR  RR       00      00
VV      VV   PP           RR  RR       00      00
  VV  VV     PP           RR    RR     00      00      ....
  VV  VV     PP           RR    RR     00      00      ....
    VV       PP           RR      RR     000000        ....
    VV       PP           RR      RR     000000        ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                                .SBTTL  $ALTLC - SET UP XPT CIRCUIT COUNTER BLOCK
                                        ;+
                                        ;
                                        ; $ALTLC - SET UP XPT CIRCUIT COUNTER BLOCK
                                        ;
                                        ; INPUTS:
                                        ;       $$HTIM - HELLO TIMER VALUE
                                        ;       $$LTIM - LISTEN TIMER VALUE
                                        ;       $$NBRA - NUMBER OF NI ROUTER ADJACENCIES
                                        ;       $$RPRI - ROUTER PRIORITY
                                        ;
                                        ; OUTPUTS:
                                        ;       CIRCUIT COUNTER BLOCK SET UP
                                        ;-

001062  012700  114224                  $ALTLC::MOV     #^RXPT,R0               ; GET XPT'S PROCESS NAME
001066                                          CALL    FNDPDV                  ; LOOK FOR XPT'S PDV
001072  103564                                  BCS     101$                    ; BR IF NOT FOUND
001074  010067  000014'                         MOV     R0,XPTPDV               ; GET PDV ADDRESS

001100  005000                                  CLR     R0                      ; ASSUME POINT-TO-POINT LINE
001102  005001                                  CLR     R1                      ; ...
001104  016704  000000G                         MOV     $SLTA,R4                ; GET SLT ADDRESS
001110  032764  000020  000000                  BIT     #LF.MTP,L.FLG(R4)       ; MULTIPOINT ?
001116  001402                                  BEQ     5$                      ; IF NOT - BRANCH

001120  116401  000014                          MOVB    L.NSTA(R4),R1           ; GET NUMBER OF STATIONS
001124  016703  000000G                 5$:     MOV     $SLTMA,R3               ; POINT TO SLT VECTOR TABLE
001130  005046                                  CLR     -(SP)                   ; INITIALIZE SYSTEM LINE NUMBER
001132  022304                          10$:    CMP     (R3)+,R4                ; IS THIS THE CORRECT SLT?
001134  001402                                  BEQ     20$                     ; BR IF YES
001136  005216                                  INC     (SP)                    ; ELSE UPDATE SYSTEM LINE NUMBER
001140  000774                                  BR      10$                     ; AND TRY AGAIN
001142  006316                          20$:    ASL     (SP)                    ; MAKE IT A WORD INDEX
001144  066716  000000G                         ADD     $LLCTA,(SP)             ; INDEX INTO REVERSE MAPPING TABLE
001150  013604                                  MOV     @(SP)+,R4               ; GET CHANNEL NUMBER AND PDV
001152  100004                                  BPL     50$                     ; BR IF NOT MULTIPOINT
001154  006304                                  ASL     R4                      ; GET STATION TABLE ADDRESS
001156  010405                                  MOV     R4,R5                   ; SAVE STATION TABLE ADDRESS
001160  060004                          40$:    ADD     R0,R4                   ; POINT TO CORRECT ENTRY IN STATION TABLE
001162  011404                                  MOV     (R4),R4                 ; GET CHANNEL NUMBER AND PDV

001164  010402                          50$:    MOV     R4,R2                   ; COPY CHANNEL NUMBER / OWNING LLC
001166  005003                                  CLR     R3                      ; INIT REG
001170  000302                                  SWAB    R2                      ; SWAP BYTE
001172  150203                                  BISB    R2,R3                   ; COPY OWNING LLC PDV INDEX
001174  016702  000000G                         MOV     $PDVTA,R2               ; COPY PDV TABLE ADDRESS
001200  060302                                  ADD     R3,R2                   ; POINT TO OWNING PDV
001202  011202                                  MOV     (R2),R2                 ; GET OWING PDV ADDRESS
001204  022762  114224  000000G                 CMP     #^RXPT,Z.NAM(R2);       XPT ?
001212  001112                                  BNE     90$                     ; IF NO - EXIT

001214  016702  000014'                         MOV     XPTPDV,R2               ; GET XPT'S PDV ADDRESS
001220  062702  000000G                         ADD     #Z.MAP,R2               ; POINT TO FORWARD MAPPING TABLE
001224  005046                                  CLR     -(SP)                   ; GET CHANNEL NUMBER
001226  110416                                  MOVB    R4,(SP)                 ; ...
001230  006316                                  ASL     (SP)                    ; MAKE IT A WORD INDEX
001232  062602                                  ADD     (SP)+,R2                ; POINT TO SLN, STA PAIR
```

```
   1137 004346  005011                          CLR     (R1)                    ; INDICATE END OF LIST
   1138 004350                                  PUTAD                           ; REWRITE COUNTER BLOCK
   1139 004354  162702  000050                  SUB     #E$NLEN,R2              ; POINT TO BEGINNING OF LAST BLOCK
   1140 004360                                  BIAS    R2                      ; SET CORRECT APR BIAS FOR ECL
   1141 004370                                  GETRV   R5,#N$ELEN              ; READ IN ECL'S DDB
   1142 004406  010267  000044G                 MOV     R2,N$ENC+2+$BFR         ; SET END OF LIST POINTER
   1143 004412                                  PUTRC                           ; WRITE UPDATED DDB
   1144
   1145 004416                          40$:    RETURN
   1146
   1147 004420                          101$:   EMSG$R  70                      ; ELSE ALLOCATION FAILURE
```

```
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05      PAGE 8
MACRO CROSS RFFERENCE                                     CREF   04.00

MACRO NAME        REFERENCES

ADJDF$            #5-59     5-69
ASL$              #5-58     7-289     7-290
ASR$              #5-59     12-633    15-936    16-1003   17-1105
BIAS              #5-59     12-647    15-947    16-1018   17-1117   17-1127   17-1140
CACDF$            #5-59     5-74
CALL               7-252    7-253     7-258     7-259     7-270     7-308     7-311     7-313     8-339     8-350
                   8-351    10-463    10-466    10-472    10-477    10-479    11-524    11-571    11-577    11-590
                   12-618   12-635    12-640    12-660    12-681    13-707    13-715    13-720    13-725    13-750
                   13-774   13-784    13-788    13-791    14-812    14-817    14-825    14-833    14-835    14-837
                   14-838   14-840    14-842    14-845    14-847    15-877    15-899    15-917    15-928    15-930
                   15-933   15-939    15-942    15-949    15-951    15-958    16-987    16-989    16-1005   16-1010
                   16-1034  16-1053   16-1065   17-1095   17-1096   17-1102   17-1107   17-1111   17-1119   17-1123
                   17-1132  17-1138   17-1141   17-1143
CALLR              7-323    10-507
CEACC$            #5-60     8-350
CRASH              7-320
CTRDF$            #5-58     5-68
DHBDF$            #5-59     5-73
ECDDB$            #5-59     5-70
EMSG$R            #5-58     7-321     10-504    10-505    11-598    12-689    13-795    14-851    15-966    16-1077
                   17-1147
ERBLK$            #7-247
ERMSG$            #7-247
ERRPT$            #7-247
GETAD             #5-58     7-308     11-577    14-838    15-951    17-1123
GETRV             #5-58     7-270     11-571    12-618    12-640    13-707    13-720    13-788    14-817    14-835
                   14-845   15-877    15-928    15-942    16-989    16-1010   16-1065   17-1096   17-1111   17-1141
NTLDF$            #5-60     5-76
PUTAD             #5-58     7-311     8-351     11-590    12-681    13-750    13-774    14-840    15-899    15-917
                   15-958   16-1053   17-1132   17-1138
PUTRC             #5-58     12-660    13-725    13-791    14-837    14-847    15-930    15-949    16-1034   17-1119
                   17-1143
RAND              #7-270    7-270     #7-308    7-308     #7-311    7-311     #8-351    8-351     #11-571   11-571
                  #11-577   11-577    #11-590   11-590    #12-618   12-618    #12-640   12-640    #12-660   12-660
                  #12-681   12-681    #13-707   13-707    #13-720   13-720    #13-725   13-725    #13-750   13-750
                  #13-774   13-774    #13-788   13-788    #13-791   13-791    #14-817   14-817    #14-835   14-835
                  #14-837   14-837    #14-838   14-838    #14-840   14-840    #14-845   14-845    #14-847   14-847
                  #15-877   15-877    #15-899   15-899    #15-917   15-917    #15-928   15-928    #15-930   15-930
                  #15-942   15-942    #15-949   15-949    #15-951   15-951    #15-958   15-958    #16-989   16-989
                  #16-1010  16-1010   #16-1034  16-1034   #16-1053  16-1053   #16-1065  16-1065   #17-1096  17-1096
                  #17-1111  17-1111   #17-1119  17-1119   #17-1123  17-1123   #17-1132  17-1132   #17-1138  17-1138
                  #17-1141  17-1141   #17-1143  17-1143
RESRG             #5-59     8-352      18-1179
RETURN             7-315    8-354      9-401    10-499    11-596    12-686    13-793    14-849    15-964    16-1074
                   17-1145  18-1180
SAVRG             #5-59     8-337     18-1162
SLTDF$            #5-58     5-72
SOB                11-583   12-679    13-748    15-897    15-915    15-957    15-961    16-1051   17-1131
SWTCHI            #5-60
SWTCHO            #5-60
VECDF$            #5-60     5-75
VNPBPT            #7-246    7-247
```

```
                                   .SBTTL  $REP.C,$REP.P - REPLACE FILE NAME
                           ;+
                           ; $REP.C - REPLACE "*" WITH CURRENT LEVEL FILE NAME
                           ; $REP.P - REPLACE "*" WITH PREVIOUS LEVEL FILE NAME
                           ;
                           ; THESE ROUTINES ARE STANDARD ERROR MESSAGE REPLACERS. $REP.C SHOULD BE USED
                           ; WHEN THE FILE IN QUESTION IS STILL OPEN, AND $REP.P SHOULD BE USED WHEN
                           ; THE FILE HAS ALREADY BEEN CLOSED.
                           ;
                           ; INPUTS:
                           ;       R3=FORMAT STRING ADDRESS
                           ;       R4=ERROR MESSAGE STRING ADDRESS
                           ;
                           ; OUTPUTS:
                           ;       R0=DESTROYED
                           ;       R3,R4=UPDATED
                           ;
                           ;-

                                   .ENABL  LSB

001222                     $REP.C::
001222  005000                     CLR     R0              ; USE CURRENT GCL LEVEL
001224  000402                     BR      10$
001226                     $REP.P::
001226  012700  000001             MOV     #1,R0           ; USE PREVIOUS GCL LEVEL
001232  112314             10$:    MOVB    (R3)+,(R4       ; LOOK FOR THE ASTERISK
001234  122724  000052             CMPB    #'*,(R4)+       ;    ...
001240  001374                     BNE     10$             ;    ...
001242  005304                     DEC     R4              ; SKIP THE ASTERISK
001244                             CALL    SLFNP           ; SETUP LOCAL FILE NAME POINTER
001250  112024             20$:    MOVB    (R0)+,(R4)+     ; COPY ASCIZ FILE NAME
001252  001376                     BNE     20$             ; INTO THE ERROR MESSAGE
001254  005304                     DEC     R4              ; SKIP THE ZERO BYTE
001256                             RETURN

                                   .DSABL  LSB
```

```
                                         .IF DF  R$$MPL                                         ;[MP01]
                               $NETPF::.WORD    0                 ; POWERFAIL ENTRY POINT       ;[MP01]
                               $NTLAL::.WORD    0                 ; CEX POOL LISTHEAD           ;[MP01]

                                       .IF DF   M$$PRO                                          ;[MP01]
                               TIMERS: .WORD    -1                ; SHOW TIMER IS NOT ACTIVE    ;[MP01]
                                       .WORD    0,0,0,0,0         ; PROCESSOR #1 ONE SEC.       ;[MP01]
                                       .WORD    0                 ; TIMER QUEUE BLOCK           ;[MP01]
                                       .WORD    0                                               ;[MP01]
                                       .WORD    0                                               ;[MP01]
                               $$$STM=.-TIMERS                                                  ;[MP01]

                                       .WORD    0,0,0,0,0         ; PROCESSOR #1 100 SMEC.      ;[MP01]
                                       .WORD    0                 ; TIMER QUEUE BLOCK           ;[MP01]
                                       .WORD    0                                               ;[MP01]
                                       .WORD    0                                               ;[MP01]
                               $$$LEN=.-TIMERS                    ; TIMER BLOCK LENGTH          ;[MP01]

                                       .REPT    M$$PRO-1          ; SETUP TIMER BLOCKS FOR OTHER ;[MP01]
                                                                  ; PROCESSORS                  ;[MP01]
                                       .WORD    -1                ; TIMER NOT ACTIVE            ;[MP01]
                                       .WORD    0,0,0,0,0         ; ONE SEC. BLOCK              ;[MP01]
                                       .WORD    0,0,0             ; TIMER QUEUE                 ;[MP01]

                                       .WORD    0,0,0,0,0         ; ONE MSEC. BLOCK             ;[MP01]
                                       .WORD    0,0,0             ; TIMER QUEUE                 ;[MP01]
                                       .ENDR                                                    ;[MP01]

                               $$$OFF=2                                                         ;[MP01]
                               $1SCTB::.REPT    M$$PRO                                          ;[MP01]
                                       .WORD    TIMERS+$$$OFF     ; POINTER TO PROCESSOR DEPENDANT;[MP01]
                                                                  ; TIMER CELL                  ;[MP01]
                               $$$OFF=$$$OFF+$$$LEN                                             ;[MP01]
                                       .ENDR                                                    ;[MP01]
                               $$$OFF=$$$STM                                                    ;[MP01]
                               $STMTB::.REPT    M$$PRO                                          ;[MP01]
                                       .WORD    TIMERS+$$$OFF     ; POINTER TO PROCESSOR DEPENDANT;[MP01]
                                                                  ; TIMER CELL                  ;[MP01]
                               $$$OFF=$$$OFF+$$$LEN                                             ;[MP01]
                                       .ENDR                                                    ;[MP01]
                               $CRESL::LCKDF$   SPIN              ; SERIALIZE ACCESS TO CEX RESOURCES ;[MP01]
                                       .ENDC    ; M$$PRO                                        ;[MP01]
                                       .ENDC    ; R$$MPL                                        ;[MP01]


004216  000000                 $SHLST::.WORD    0                 ; SECONDARY HOST LISTHEAD
004220  000000  000000         SPARE1::.WORD    0,0               ; FIRST DOUBLE WORD SPARE CELL
004224  000000  000000         SPARE2::.WORD    0,0               ; SECOND DOUBLE WORD SPARE CELL

004230  177777  177776                 .WORD    -1,-2             ; FF ( FUDGE FACTOR )
004234                         $CEXND::                           ; END OF CEXCM

000000                                 .PSECT   CET$$A,RW,OVR     ; START OF EXTENSIBLE PROGRAM SECTION
000000                         $CETAB::                           ; REFERENCE LABEL
000000                                 .PSECT   CET$$B,RW,CON     ; FORCE ALPHABETICAL REFERENCE
000000                         $CETND::                           ; REFERENCE LABEL
004234                                 .PSECT
```

```

                                .IF DF  R$$MPL                                                          ;[MP01]
                                .SBTTL  DBLK - CALCULATE D-SPACE OFFSETS                                ;[MP01]
                        ;+                                                                              ;[MP01]
                        ; DBLK - CALCULATE D-SPACE BLOCK AND OFFSET                                     ;[MP01]
                        ;                                                                               ;[MP01]
                        ; INPUTS:                                                                       ;[MP01]
                        ;       $RBLCK - START OF D-SPACE                                               ;[MP01]
                        ;       $RADDR - VIRTUAL ADDRESS                                                ;[MP01]
                        ;                                                                               ;[MP01]
                        ; OUTPUTS:                                                                      ;[MP01]
                        ;       $RBLCK, $RADDR ADJUSTED FOR FILE ACCESS ROUTINES                        ;[MP01]
                        ;-                                                                              ;[MP01]
                        DBLK:   CALL    $SAVAL          ; SAVE REG'S                                    ;[MP01]
                                SUB     #20000,$RADDR   ; REMOVE FIST APR BLOCK                         ;[MP01]
                        10$:    SUB     #20000,$RADDR   ; ADJUST VIRTUAL ADDRESS                        ;[MP01]
                                BCS     20$             ; IF LESS THAN 4K LEFT - BRANCH RIGHT           ;[MP01]
                                ADD     #200,$RBLCK     ; COUNT ANOTHER 4K BIAS                         ;[MP01]
                                BR      10$             ; CONTINUE                                      ;[MP01]

                        20$:    ADD     #20000,$RADDR   ; REGAIN VIRTUAL OFFSET INTO LAST BLOCK         ;[MP01]
                        30$:    CLC                     ; CLEAR C-BIT                                   ;[MP01]

                                RETURN                                                                  ;[MP01]

                                .ENDC                                                                   ;[MP01]

        000001                  .END
```

```
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 9
SYMBOL  CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

ZF.TIM   = 000200   G       #5-79
ZF.X3P   = 000000   G       #5-79
ZS.ASN   = 100000   G       #5-79
ZS.BSY   = 140000   G       #5-79
Z.AVL      000014   G       #5-79
Z.DAT      000016   G       #5-79
Z.DSP      000000   G       #5-79         5-79
Z.FLG      000010   G       #5-79
Z.LEN    = 000016   G       #5-79
Z.LLN      000006   G       #5-79
Z.MAP      000020   G       #5-79
Z.NAM      000004   G       #5-79
Z.PCB      000012   G       #5-79
Z.SCH      000007   G       #5-79
$ADRX      000504  RG       #6-134
$ADRZ      000502  RG       #6-133
$ASTLV     000210  RG       #6-102
$AZFS      001042  RG       #8-232
$BFR       000000  RG        6-112      #7-204     *19-1136     23-1285
$BIAS      002346  RG       #15-517
$BIGCT     000726  RG       #7-171
$CBDSG   = ******   GX       11-353
$CCBAF     003734  RG       #17-773
$CCBAL     003740  RG       #17-775
$CCBCT     003732  RG       #17-772
$CCRLH     004040  RG       #17-816
$CCBNM     003716  RG       #17-766
$CCBSZ     003720  RG       #17-767
$CEACC     005622  RG        6-127      #24-1312
$CEAVL     004236  RG       #17-929
$CELFN     004210  R        #17-867
$CET       001564  RG       #13-393
$CETAB     000000  RG       #17-923
$CETDF     004242  RG       #17-932
$CETLN     004244  RG       #17-934
$CETND     000000  RG       #17-925
$CETST     004246  RG       #17-935
$CEX       001602  RG       #13-395
$CEXND     004234  RG       #17-920
$CLASN     000162  RG       #6-102
$CLAST     000172  RG       #6-102
$CLBUF   = 000310  RG       #6-102        14-431      14-438       22-1256
$CLCSI     000212  RG       #6-102        6-102       6-113
$CLEVL     000206  RG       #6-102        12-370
$CLFDB     000266  RG       #6-102        6-102       6-102
$CLFLG     000464  RG       #6-102
$CLIOS   = 000132  RG       #6-102        11-351      22-1270
$CLMCR     000000  RG       #6-102        6-102
$CLPMT     000466  RG       #6-102
$CLPRM     000472  R         6-102       #6-102       6-102
$CLQIO     000122  RG       #6-102        6-102
$CLRTN   = ******   GX        6-102
```

```
    125 000126  001544' 000000                  .WORD   SLIN,0
    126 000132  002536' 000000          .CCEX:  .WORD   CCEX,0          ; CHECK CLEAR SYSTEM REQUEST
    127 000136  000176'                 .CPRO:  .WORD   CEXLD           ; CEX LOADED ?
    128 000140  000474'                         .WORD   PDVA            ; FIND PDV ADDRESS
    129 000142  001672' 000000                  .WORD   CPRO,0
    130 000146  000176'                 .CLIN:  .WORD   CEXLD           ; COMEXEC MUST BE LOADED
    131 000150  000774'                         .WORD   SLTA            ; GET SLT ADDRESS
    132 000152  002472' 000000                  .WORD   CLIN,0
    133 000156  000120' 000000          .NOP:   .WORD   NOP,0           ; NO CHECKING REQUIRED
    134 000162                          .SNOD:
    135 000162  000176'                 .CNOD:  .WORD   CEXLD           ; COMEXEC MUST BE LOADED
    136 000164  000234' 000000                  .WORD   CHKNOD,0        ; CHECK NODE PARAMETERS
    137 000170                          .SLOG:
    138 000170  000176'                 .CLOG:  .WORD   CEXLD           ; COMEXEC MUST BE LOADED
    139 000172  000746' 000000                  .WORD   LLIN,0          ; CHECK FOR LINE
    140 000176  000176'                 .DMOD:  .WORD   CEXLD           ; COMMEXEC MUST BE LOADED
    141 000200  000216' 000000                  .WORD   PSILD,0         ; PSI MUST BE LOADED
    142 000204  000176'                 .SCIR:  .WORD   CEXLD           ; MAKE SURE COMM EXEC IS LOADED
    143 000206  000774'                         .WORD   SLTA            ; FIND SLT ADDRESS
    144 000210  001532'                         .WORD   CLRPP           ; CLEAR PROCESS PARAMETERS
    145 000212  001640' 000000                  .WORD   SCIR,0
    146 000216  000176'                 .SMOD:  .WORD   CEXLD           ; MAKE SURE COMM EXEC IS LOADED
    147 000220  000216'                         .WORD   PSILD           ; PSI MUST BE LOADED
    148 000222  000000                          .WORD   0
    149 000224                          .SALI:
    150 000224                          .CALI:
    151 000224                          .SOBJ:
    152 000224                          .COBJ:
    153 000224                          .DCEX:
    154 000224                          .DLIN:
    155 000224                          .DPRO:
    156 000224                          .DNOD:
    157 000224                          .DALI:
    158 000224                          .DLOG:
    159 000224                          .DCIR:
    160 000224  000176' 000000          .DOBJ:  .WORD   CEXLD,0         ; COMEXEC MUST BE LOADED
    161 000230  000162' 000000          .RNTI:  .WORD   RNTI,0          ; COMEXEC MUST NOT BE LOADED
```

```
                                                .SBTTL  LLIN - CHECK FOR LINE IN SYSTEM FOR LOGGING

                                        ;+
                                        ;
                                        ; LLIN - CHECK FOR LINE IN SYSTEM FOR EVENT LOGGING AND RETURN
                                        ;        SYSTEM LINE NUMBER IF IT EXISTS
                                        ;
                                        ; INPUTS:
                                        ;       RQB - REQUEST BLOCK FOR SET OR CLEAR LOGGING COMMAND
                                        ;
                                        ; OUTPUTS:
                                        ;       $SLN - SYSTEM LINE NUMBER
                                        ;
                                        ;-

000746  032767  000040  000000G LLIN:   BIT     #OP$LNE,$OPTON          ; WAS A LINE SPECIFIED?
000754  001406                          BEQ     10$                     ; BR IF NO
000756  010346                          MOV     R3,-(SP)                ; SAVE R3
000760  012703  000030G                 MOV     #RQB+LO.LID,R3          ; POINT TO DEVICE NAME IN REQUEST BLOCK
000764                                  CALL    SLTA                    ; GET SLN
000770  012603                          MOV     (SP)+,R3                ; RESTORE R3
000772                          10$:    RETURN
```

```
   993
   994 002652                               TSKSCH:
   995 002652  004567  000000G                      JSR     R5,$SAVRG          ; SAVE NON-VOLATILE REGISTERS
   996 002656                                       GETRV   .TSKHD,.TLGTH      ; READ IN LISTHEAD
   997 002676  012705  000000G                      MOV     #$BFR,R5           ; POINT AT BUFFER BASE
   998 002702  011565  000030                       MOV     (R5),T.TCBL(R5)    ; INITIALIZE FOR LOOP
   999
  1000 002706  016567  000030  000000G 10$:         MOV     T.TCBL(R5),$RADDR  ; SET TCB ADDRESS
  1001 002714  000261                               SEC                        ; ASSUME ERROR
  1002 002716  001411                               BEQ     30$                ; NULL TASK ( END OF LIST )
  1003 002720                                       GETRV                      ; READ IN TCB
  1004 002724  021065  000006                       CMP     (R0),T.NAM(R5)     ; MUST MATCH BOTH
  1005 002730  001366                               BNE     10$
  1006 002732  026065  000002  000010               CMP     2(R0),T.NAM+2(R5)  ; HALVES OF NAME
  1007 002740  001362                               BNE     10$                ; C-BIT IS CLEAR
  1008 002742                          30$:         RETURN
```

```
SYMBOL  VALUE             REFERENCES

E.MON     012716 R        #79-3491
E.MSK     012634 R        #78-3463
E.NUM     012554 R        #77-3438
E1LEN   = 000031          #10-629         10-629
E10LEN  = 000040          #10-636         10-636
E11LEN  = 000046          #10-637         10-637
E12LEN  = 000046          #10-638         10-638
E13LEN  = 000035          #10-639         10-639
E14LEN  = 000042          #10-640         10-640
E15LEN  = 000034          #10-641         10-641
E16LEN  = 000034          #10-642         10-642
E17LEN  = 000030          #10-643         10-643
E2LEN   = 000062          #10-630         10-630
E20LEN  = 000034          #10-644         10-644
E21LEN  = 000056          #10-645         10-645
E22LEN  = 000040          #10-646         10-646
E23LEN  = 000055          #10-647         10-647
E24LEN  = 000034          #10-648         10-648
E25LEN  = 000031          #10-649         10-649
E26LEN  = 000022          #10-650         10-650
E27LEN  = 000033          #10-651         10-651
E3LEN   = 000050          #10-631         10-631
E30LEN  = 000012          #10-652         10-652
E31LEN  = 000034          #10-653         10-653
E32LEN  = 000041          #10-654         10-654
E33LEN  = 000037          #10-655         10-655
E34LEN  = 000037          #10-656         10-656
E35LEN  = 000036          #10-657         10-657
E36LEN  = 000032          #10-658         10-658
E37LEN  = 000032          #10-659         10-659
E4LEN   = 000044          #10-632         10-632
E40LEN  = 000034          #10-660         10-660
E41LEN  = 000044          #10-661         10-661
E42LEN  = 000034          #10-662         10-662
E43LEN  = 000034          #10-663         10-663
E44LEN  = 000036          #10-664         10-664
E45LEN  = 000037          #10-665         10-665
E5LEN   = 000034          #10-633         10-633
E6LEN   = 000051          #10-634         10-634
E7LEN   = 000046          #10-635         10-635
FE.CEX  = 020000            8-225         16-939
FE.PLA  = 000020            8-327
FF.ADD  = 000020          #5-77           5-77
FF.CIR  = 000040          #5-77           5-77
FF.CON  = 000001          #5-77           79-3487
FF.FIL  = 000002          #5-77           79-3489
FF.HST  = 040000          #5-77
FF.LIN  = 000010          #5-77           5-77
FF.MOD  = 000100          #5-77           5-77
FF.MON  = 000004          #5-77           79-3491
FF.MSK  = 000077          #5-77
FF.PRT  = 000200          #5-77           5-77
```

```
                                                        H  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 17
SYMBOL  CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE            REFERENCES

$ALL16   = ******  GX       12-786
$ALL22   = ******  GX        8-492
$ALPHA   = 000022         #18-1108  #19-1139  #20-1178  #21-1220  #22-1250  #23-1265  #32-1884  #46-2329
$ALPOL   = ******  GX      64-2887   69-3163   79-3494   86-3675   94-4026  102-4433
$ANY     = 000020         #18-1108  #19-1139  #20-1178  #21-1220  #22-1250  #23-1265  #32-1884  #46-2329
$ASUMR   = ******  GX       11-730
$AZFS    = ******  GX      29-1729
$BFR     = ******  GX       *8-266     8-268     8-271     8-273     8-275     8-278     8-281     8-284     8-288
                             8-291     8-294     8-297    *8-297     8-299    *8-331    *8-333    *8-336     8-373
                            *8-400    *8-401    *8-416    *8-417    *8-446    *8-447    *8-458    11-743   *12-795
                           *12-796   *13-830   *13-831   *14-875   *14-876    16-949  *17-1042   17-1052   17-1057
                           17-1063   17-1065   17-1070   17-1072   17-1081   17-1082   26-1538  *26-1549  *26-1550
                          *26-1556  *26-1557  *26-1558  *26-1559  *26-1566   26-1574   26-1575   26-1578  *26-1584
                          *26-1586  *26-1590   26-1595   26-1598   26-1599  *26-1608  *26-1610  *26-1615   61-2754
                           61-2763   61-2777  *62-2796  *62-2818  *63-2842  *63-2865   64-2890   64-2895   65-2922
                           65-2923   65-2928   65-2961  *67-3018  *67-3028  *67-3038  *67-3047  *67-3057  *67-3066
                          *67-3075  *67-3084  *67-3094  *67-3104  *67-3111   69-3166   69-3176   69-3185   69-3192
                          *70-3241  *71-3259  *72-3282  *72-3283  *72-3284  *73-3304   74-3320  *74-3328   75-3346
                          *75-3355  *75-3374  *76-3394   76-3410  *76-3418   79-3496  *80-3535   80-3539   80-3543
                           86-3678   86-3691  *87-3720   87-3723   87-3727   88-3748  *88-3751   88-3755   88-3763
                           94-4035  *94-4041  *94-4042  *94-4043  *94-4045  *94-4046  *94-4047  *94-4048  *94-4050
                          *94-4051  *94-4052  *94-4054  *94-4055  *94-4056  *94-4057  *94-4058   94-4064   94-4077
                           98-4259   98-4273   98-4274  102-4442  102-4455  102-4468 *103-4525 *103-4531  103-4532
                         *105-4708
$BLANK   = 000006         #18-1108  #19-1139  #20-1178  #21-1220  #22-1250  #23-1265  #32-1884  #46-2329
$CAT5    = ******  GX      24-1478   24-1483   55-2624   83-3606   83-3610   91-3907   92-3937
$CBDSG   = ******  GX       10-600
$CCBCT   = ******  GX     *54-2607
$CCBLH   = ******  GX       12-795   *12-798    16-946   *16-949
$CCBNM   = ******  GX       12-776   53-2585  *54-2606   54-2607
$CCBS      002554 RG         8-472   #12-775
$CCBSZ   = ******  GX       12-778    12-797    16-944   53-2586
$CEACC   = ******  GX      65-2925   65-2955   69-3174   79-3505   80-3541   86-3685   87-3725   88-3761   94-4033
                           98-4271  102-4450  103-4523  103-4529  105-4706

$CET     = ******  GX      29-1727
$CFG       005410 RG         8-452  #28-1682
$CLBUF   = ******  GX        9-557   30-1844   31-1871
$CLDEQ   = ******  GX      28-1707   30-1790   30-1805   30-1819   30-1830
$CLFLG   = ******  GX     *28-1684
$CLIOS   = ******  GX      28-1710   30-1754   31-1867
$CLQUE   = ******  GX      29-1730
$CLSAV   = ******  GX        8-221   28-1683
$CMAND   = ******  GX    *107-4779 *108-4817 *109-4851
$CRPAR     003366 RG         8-317     8-483  #17-1041
$CTIM    = ******  GX     *60-2725
$CXOPT   = ******  GX      54-2604
$DALCT     003200 RG         9-541   #16-938
$DEA16   = ******  GX       16-950   16-1020   26-1630
$DECPT   = ******  GX        8-457   61-2753   62-2792   62-2808   63-2841   63-2863   64-2889   65-2920   65-2960
                           67-3017   67-3027   67-3037   67-3046   67-3056   67-3065   67-3074   67-3083   67-3093
                           67-3103   67-3110   69-3184   69-3191   70-3239   71-3257   72-3280   73-3303   74-3318
                           75-3344   75-3372   76-3392   76-3408   88-3747   94-4063   94-4076   98-4258  102-4454
```

```
                                                        I  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 18
```

```
     92                                          .SBTTL  CONSTANT DEFINITIONS
     93                                  ;***
     94                                  ; LOCAL SYMBOLS
     95                                  ;***
     96
     97                                  ;
     98                                  ; ERROR CODES RETURNED TO CALLER
     99                                  ;
    100 000050                                   .ASECT
    101                 000002                   .=2                             ; 0=SUCCESS
    102 000002                           ERR2:   .BLKW   1                       ; OPEN FAILURE
    103 000004                           ERR4:   .BLKW   1                       ; GET ATTRIBUTES FAILED
    104 000006                           ERR6:   .BLKW   1                       ; FILE NOT CONTIGUOUS
    105 000010                           ERR8:   .BLKW   1                       ; LABEL BLOCK READ FAILURE
    106 000012                           ERR10:  .BLKW   1                       ; INVALID PROCESS IMAGE
    107 000014                           ERR12:  .BLKW   1                       ; PROCESS NOT BUILT FOR APR5 (MAPPED)
    108 000016                           ERR14:  .BLKW                           ; PROCESS > 4K WORDS (MAPPED)
    109 000020                           ERR16:  .BLKW   1                       ; PROCESS EXTENSION > 4K WORDS (MAPPED)
    110 000022                           ERR18:  .BLKW   1                       ; PARTITION NOT IN SYSTEM
    111 000024                           ERR20:  .BLKW   1                       ; PROCESS/PARTITION BASE MISMATCH
    112 000026                           ERR22:  .BLKW   1                       ; PARTITION TOO SMALL
    113 000030                           ERR24:  .BLKW   1                       ; NOT COMMON PARTITION
    114 000032                           ERR26:  .BLKW   1                       ; PARTITION BUSY
    115 000034                           ERR28:  .BLKW   1                       ; CEX PARTITION NOT IN EXEC SPACE (MAPPED)
    116 000036                           ERR30:  .BLKW   1                       ; SUB-PCB ALLOCATION FAILURE (MAPPED)
    117 000040                           ERR32:  .BLKW   1                       ; MAIN PARTITION TOO FRAGMENTED (MAPPED)
    118 000042                           ERR34:  .BLKW   1                       ; PROCESS READ FAILURE
    119 000000                                   .PSECT
```

```
                                        .SBTTL  PR.PAR - FIND PARTITION
                                ;+
                                ; PR.PAR - FIND PARTITION
                                ;
                                ; THIS ROUTINE IS CALLED TO FIND THE PROCESS PARTITION. IF THE PARTITION IS A
                                ; SYSTEM CONTROLLED PARTITION, A SUB-PARTITION PCB WILL BE CREATED.
                                ;
                                ; INPUTS:
                                ;       R0=LABEL BLOCK BUFFER ADDRESS
                                ;       $$PAR=PARTITION NAME (OPTIONAL)
                                ;
                                ; OUTPUTS:
                                ;       C-BIT=SUCCESS/FAILURE
                                ;       $PNAME=PARTITION NAME
                                ;       PCBA=PROCESS PCB ADDRESS
                                ;-
001540  042767  000000G 000000G PR.PAR: BIC     #FL.TOP,$FLAGS  ; TURN OFF TOP-DOWN LOADING

001546  032767  000001  000150'         BIT     #MCLOA,MCFLG    ; MICROCODE PARTITION ?
001554  001407                          BEQ     10$             ; IF NO - BRANCH

001556  012767  026226  000254'         MOV     #^RGEN,$PNAME   ; ALWAYS USE GEN
001564  012767  000000  000256'         MOV     #^R   ,$PNAME+2 ; ...
001572  000415                          BR      20$             ; CONTINUE

001574  016767  000002G 000256' 10$:    MOV     $$PAR+2,$PNAME+2 ; COPY PARTITION OVERRIDE NAME
001602  016767  000000G 000254'         MOV     $$PAR,$PNAME    ; ...
001610  001006                          BNE     20$             ; IF NE, USE OVERRIDE NAME
001612  016067  000000G 000254'         MOV     L$BPAR(R0),$PNAME ; ELSE USE PARTITION FROM PROCESS FILE
001620  016067  000002G 000256'         MOV     L$BPAR+2(R0),$PNAME+2 ; ...
001626  016767  000000G 000000G 20$:    MOV     $PARHD,$BFR     ; GET POINTER TO FIRST PCB
001634  016705  000000G         30$:    MOV     $BFR,R5         ; GET NEXT PCB
001640  001002                          BNE     40$             ; IF NON-ZERO .. MORE IN LIST
001642  000167  000406                  JMP     120$            ; IF ZERO, END OF LIST
001646                          40$:    GETRV   R5,,PLGTH       ; READ PCB INTO BUFFER
001664  026767  000254' 000004G         CMP     $PNAME,P.NAM+$BFR ; LOOK FOR MATCH BETWEEN $PNAME
001672  001360                          BNE     30$             ; AND PARTITION NAME FROM PCB
001674  026767  000256' 000006G         CMP     $PNAME+2,P.NAM+2+$BFR ; ...
001702  001354                          BNE     30$             ;       ...

                                        .IF DF  R$$MPL                                          ;[MP01]

                                        CALL    PR.SUB                                          ;[MP01]

                                        .IFF

001704  032767  000040  000030G         BIT     #PS.SYS,P.STAT+$BFR ; SYSTEM CONTROLLED PARTITION?
001712  001446                          BEQ     60$             ; IF EQ, NO
001714  032767  000000G 000000G         BIT     #FL.CEX,$FLAGS  ; IS THIS THE COMM EXEC?
001722  001170                          BNE     150$            ; IF NE, YES

                                        .IF NDF R$$MPL                                          ;[MP01]
001724  032767  000000G 000000G         BIT     #FL.TOP,$FLAGS  ; LOAD TOP ?
001732  001433                          BEQ     50$             ; IF NO - BRANCH
                                        .ENDC                                                   ;[MP01]

001734  016701  000014G                 MOV     $BFR+P.REL,R1   ; GET START OF PARTITION
```

```
                                                                     H  5
VPIO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:04      PAGE 4
SYMBOL  CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE            REFERENCES

$MUL     = ******  GX     12-517     12-530     13-606
$NAME    = ******  GX     10-341     11-425     11-429     12-500     12-555     13-597     15-785     15-790
$PARHD   = ******  GX     10-328     14-646
$PDVNM   = ******  GX     17-933
$PDVTA   = ******  GX     17-932     17-945
$PNAME     000254  RG     #7-177    *14-637    *14-638    *14-641    *14-642    *14-644    *14-645     14-651     14-653
$PRCLN   = ******  GX     12-529
$PREXT   = ******  GX     12-525
$PRINC   = ******  GX     12-512
$PRIO      000000  RG     #9-265
$PRLLN   = ******  GX     12-516
$PUTRC   = ******  GX     9-295      10-386     14-724     14-728     15-805     15-822     15-826     15-854     15-859
$RADDR   = ******  GX     *9-292    *10-333    *10-383    *13-599    *14-650    *14-719    *14-722    *14-726    *15-805
                          *15-809   *15-813    *15-819    *15-828    *15-831    *15-851    *15-857    *16-901
$READX   = ******  GX     16-904
$RLBL    = ******  GX     11-455     12-489
$SAVAL   = ******  GX     17-960
$TIOUT   = ******  GX     17-963
$VFLAG   = ******  GX     10-325     14-676     15-834
$XPAR    = ******  GX     9-283      10-372     10-374
$$       = 177777         7-133      7-133      #7-133     #7-134     7-136      7-136      #7-136     #7-137
$$EXT    = ******  GX     12-528     12-537
$$INC    = ******  GX     12-515
$$PAR    = ******  GX     14-641     14-642
$$$      = 000054  R      #7-133     7-134      #7-136     7-137
$$$$     = 000136  R      #7-134     #7-137
.PLGTH   = ******  GX     9-292      10-333     10-383     14-650     15-772     15-777     15-805     15-869     16-901
.PPRO    = ******  GX     9-293      10-384
```

```
VPRO - VNP SET PROCESS ROUTINES MACRO V05.03b Wednesday 28-Aug-85 20:04
Table of contents
```

```
     5-   54     MACRO DEFINITIONS
     6-   80     LOCAL DATA
     8-  326     ALVBLK - ALLOCATE VECTOR BLOCK
     9-  357     AXSYM - AUX SPECIAL SYMBOLS
    11-  509     $ALTLC - SET UP XPT CIRCUIT COUNTER BLOCK
    12-  600     $ROL - ALLOCATE REACHABILITY TABLES
    13-  693     $CACH - ALLOCATE CACHE
    14-  799     $ALADJ  - ALLOCATE ADJACENCY DATA BASE
    15-  854     $CKT - ALLOCATE XPT CIRCUIT COUNTERS
    16-  970     $ALMHC - ALLOCATE MINHOPS/MINCOST MATRICES
    17- 1080     $ALECL - ALLOCATE ECL NODE COUNTERS
    18- 1149     FNDPDV - FIND A PDV ADDRESS
    19- 1183     $CTSYM - FIND SYMBOLS FROM CETAB                        ;[MP01]
```

```
001234  011204                                  MOV     (R2),R4            ; GET SLN, STA PAIR
001236  016702  000014'                         MOV     XPTPDV,R2          ; GET XPT'S PDV ADDRESS

001242  016267  000000G 000016'                 MOV     Z.DSP(R2),XPTDSP ; SAVE XPT'S MAPPING BIAS
001250  016202  000000G                         MOV     Z.DAT(R2),R2       ; GET ADDRESS OF XPT'S DDB
001254                                          GETRV   R2,#N$XLEN         ; READ XPT'S DDB

001272  016703  000100G                         MOV     N$TLC+$BFR,R3      ; GET NUMBER OF COUNTERS
001276  016702  000104G                         MOV     N$TLC+4+$BFR,R2 ; GET ADDRESS OF FIRST COUNTER BLOCK

001302  016767  000102G 000000G                 MOV     N$TLC+2+$BFR,$RBLCK ; MAP TO XPT'S PROCESS SPACE
001310                                  60$:    GETAD   R2,#T$LLEN         ; READ IN COUNTER BLOCK
001326  005767  000000G                         TST     $BFR               ; HAS THIS BLOCK BEEN ASSIGNED ALREADY?
001332  100410                                  BMI     70$                ; BR IF NO - USE IT
001334  020467  000000G                         CMP     R4,$BFR            ; ELSE IS THIS THE CORRECT ENTRY?
001340  001407                                  BEQ     80$                ; BR IF YES
001342  062702  000046                          ADD     #T$LLEN,R2         ; POINT TO NEXT COUNTER BLOCK
001346  077                                     SOB     R3,60$             ; CHECK ALL COUNTER BLOCKS
001352  000432                                  BR      90$                ; CONTINUE
001354  010467  000000G                 70$:    MOV     R4,$BFR            ; STORE CHANNEL NUMBER AND PDV
001360  016067  000000G 000030G         80$:    MOV     $$HTIM(R0),T$T5+$BFR ; STORE HELLO TIMER
001366  016067  000000G 000032G                 MOV     $$LTIM(R0),T$T6+$BFR ; STORE LISTEN TIMER
001374  016767  000000G 000042G                 MOV     $$NBRA,T$NRNI+$BFR ; STORE NI ROUTER ADJACENCY COUNT
001402  016767  000000G 000040G                 MOV     $$RPRI,T$RPRI+$BFR ; STORE ROUTER PRIORITY
001410                                          PUTAD   R2,#T$LLEN         ; WRITE UPDATED COUNTER BLOCK
001426  062700  000002                          ADD     #2,R0              ; UPDATE OFFSET FOR NEXT STATION ENTRY
001432  010504                                  MOV     R5,R4              ; RESTORE STATION TABLE ADDRESS
001434  005301                                  DEC     R1                 ; MORE STATIONS TO CHECK?
001436  003250                                  BGT     40$
001440  000241                          90$:    CLC
001442                                          RETURN
                                        ;
001444                                  101$:   EMSG$R  70                 ; XPT NOT FOUND???
```

```
                                                .SBTTL  FNDPDV - FIND A PDV ADDRESS
                                        ;+
                                        ; FNDPDV - TRY TO FIND PDV
                                        ;
                                        ; INPUTS:
                                        ;       R0=RAD50 PDV NAME
                                        ;
                                        ; OUTPUTS:
                                        ;       C-BIT=SUCCESS/FAILURE
                                        ;       R0=PDV ADDRESS
                                        ;       R1=PDV INDEX
                                        ;-
004426                                  FNDPDV:
004426                                          SAVRG   <R2>                    ; SAVE R2
004430  010046                                  MOV     R0,-(SP)                ; SAVE NAME TO BE MATCHED
004432  016701  000000G                         MOV     $PDVTA,R1               ; GET PDV TABLE ADDRESS
004436  016702  000000G                         MOV     $PDVNM,R2               ; AND NUMBER OF PDV'S
004442  001407                                  BEQ     30$                     ; IF ZERO, NO PDV'S!
004444  012100                          10$:    MOV     (R1)+,R0                ; GET PDV ADDRESS
004446  001403                                  BEQ     20$                     ; IF ZERO, SKIP THIS ONE
004450  026016  000000G                 15$:    CMP     Z.NAM(R0),(SP)          ; DOES THE NAME MATCH?
004454  001403                                  BEQ     40$                     ; IF EQ, YES (C-BIT CLEAR)
004456  005302                          20$:    DEC     R2
004460  001371                                  BNE     10$                     ; LOOP $PDVNM TIMES
004462  000261                          30$:    SEC                             ; INDICATE FAILURE
004464  005226                          40$:    INC     (SP)+                   ; PURGE STACK
004466  006016                                  ROR     (SP)                    ; SAVE C-BIT
004470  005741                                  TST     -(R1)                   ; CALCULATE PDV INDEX
004472  166701  000000G                         SUB     $PDVTA,R1               ;   ...
004476  006316                                  ASL     (SP)                    ; RESTORE C-BIT
004500                                          RESRG   <R2>                    ; RESTORE R2
004502                                          RETURN
```

```
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05     PAGE 9
MACRO CROSS REFERENCE                                    CREF   04.00
MACRO NAME       REFERENCES
XPDDB$           #5-58       5-71
```

```
                                        .SBTTL  $REP.N - REPLACE ERROR CODE
                                        .SBTTL  $CUR.N,$PRV.N - REPLACE FILE NAME AND ERROR CODE
                                ;+
                                ; $REP.N - REPLACE "*" WITH @$CLIOS (ERROR CODE) CONVERTED TO DECIMAL
                                ; $CUR.N - REPLACE CURRENT LEVEL FILE NAME AND ERROR CODE
                                ; $PRV.N - REPLACE PREVIOUS LEVEL FILE NAME AND ERROR CODE
                                ;
                                ; INPUTS:
                                ;       R3=FORMAT STRING ADDRESS
                                ;       R4=ERROR MESSAGE STRING ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       R0,R1,R2=DESTROYED
                                ;       R3,R4=UPDATED
                                ;-
001260                          $CUR.N::
001260                                  CALL    $REP.C          ; REPLACE CURRENT LEVEL FILE NAME
001264  000402                          BR      $REP.N
001266                          $PRV.N::
001266                                  CALL    $REP.P          ; REPLACE PREVIOUS LEVEL FILE NAME
001272                          $REP.N::
001272  010500                          MOV     R5,R0           ; COPY ERROR MESSAGE BLOCK ADDRESS
001274  005720                          TST     (R0)+           ; NOW POINTS AT TEXT STRING
001276  122720  000050          10$:    CMPB    #'(,(R0)+       ; LOOK FOR LEFT PAREN
001302  001375                          BNE     10$             ;   ...
001304  117701  000132'                 MOVB    @$CLIOS,R1      ; GET ERROR CODE
001310  005002                          CLR     R2              ; SUPPRESS LEADING ZEROES
001312                                  CALL    $CBDSG          ; CONVERT TO SIGNED DECIMAL
001316  112720  000056                  MOVB    #'.,(R0)+       ; DEPOSIT DECIMAL POINT
001322  112720  000051                  MOVB    #'),(R0)+       ; RIGHT PAREN
001326  105010                          CLRB    (R0)            ; AND ZERO BYTE (END OF TEXT INDICATOR)
001330                                  RETURN
```

```

004234  000000                  .WORD   0
004236  000000          $CEAVL::.WORD   0                       ; CEX POOL LISTHEAD IMAGE
004240  000000                  .WORD   0

004242  000000          $CETDF::.WORD   0                       ; SHIFT DIFFERENTIAL BETWEEN BUFFER AND
                                                                ;  REAL ADDRESSES IN CETAB
004244  000000          $CETLN::.WORD   0                       ; LENGTH OF CETAB IMAGE
004246  000000          $CETST::.WORD   0                       ; ACTUAL START OF CETAB
004250  000000                  .WORD   0
004252  000000          $CRAVL::.WORD   0                       ; RSX POOL LISTHEAD IMAGE
004254  000000                  .WORD   0

004256  000003                  .WORD   3                       ; ROUNDING FACTOR
004260  000000          $PRAVL::.WORD   0                       ; OFFSET TO FREE SPACE LISTHEAD
004262  000000                  .WORD   0

                                .IF DF  R$$MPL                                                  ;[MP01]
                        $POLHD::.WORD   0                       ; SECONDARY POOL LISTHEAD       ;[MP01]
                                .WORD   0                       ; AVAILABLE BLOCKS              ;[MP01]

                                .WORD   3                       ; ICB POOL ROUNDING FACTOR      ;[MP01]
                        $ICAVL::.WORD   0                       ; ICB POOL LISTHEAD             ;[MP01]
                                .WORD   0                       ; FIRST BLOCK SIZE              ;[MP01]
                                .WORD   0                       ; DUMMY WORD                    ;[MP01]

                                .IF DF  M$$PRO                                                  ;[MP01]
                                .REPT   <M$$PRO-1>              ; SET UP ICB POOL LISTHEADS     ;[MP01]
                                                                ; FOR REMAINING PROCESSORS      ;[MP01]
                                .WORD   3                       ; ICB POOL ROUNDING FACTOR      ;[MP01]
                                .WORD   0                       ; ICB POOL LISTHEAD             ;[MP01]
                                .WORD   0                       ; FIRST BLOCK SIZE              ;[MP01]
                                .WORD   0                       ; DUMMY WORD                    ;[MP01]
                                .ENDR                                                           ;[MP01]

                                .ENDC                                                           ;[MP01]

                                .ENDC                                                           ;[MP01]

004264  000000          $NTUIC::.WORD   0                       ; NETWORK UIC

004266                  $NTLHB::.BLKW   32.                     ; HOME BLOCK IMAGE (1ST BLOCK OF NTPOOL)

```

```
                                                                       H 12
VRES - VNP ROOT MACRO V05.03b Tuesday 17-Sep-85 16:15  Page 26-1
Symbol table

ARAMSK= 176000 G        D$AMXH  000074        FE.MXT= 040000        F.FVER= 000120        G$$TSS= 000000
A$$CHK= 000000          D$ANN   000000        FE.NLG= 100000        F.HIBK= 000004        G$$TTK= 000000
A$$CPS= 000000          D$BRPR  000102        FE.OFF= 001000        F.LUN = 000042        G$$WRD= 000000
A$$PRI= 000000          D$BRTM  000100        FE.PKT= 000100        F.MBCT= 000054        G.MCRB= 000002 G
A$$TRP= 000000          D$DELF  000045        FE.PLA= 000020        F.MBC1= 000055        HF.CIS= 000200
A.LULU= 000002          D$DELW  000046        FE.X25= 004000        F.MBFG= 000056        HF.EIS= 000002
A.LUNA= 000004          D$END = 000104        FILE    001374RG      F.NRBD= 000024        HF.FPP= 100000
A.LUNU= 000006          D$FNB   000034        FL.CEX= 000010 G      F.NREC= 000030        HF.QB = 000004
CEACC   000500RG        D$HIOR  000024        FL.CHA= 000020 G      F.OVBS= 000030        HF.UBM= 000001
CFSFLG  001026RG        D$HOST  000022        FL.CNT= 004000 G      F.RACC= 000016        IE.EOF= ****** GX
CL.AST= 004000          D$INAC  000044        FL.DDM= 000001 G      F.RATT= 000001        IO.WLB= ****** GX
CL.ATT= 000200          D$INCT  000042        FL.DLC= 000002 G      F.RCNM= 000034        I$$RAR= 000000
CL.CAS= 020000          D$IPL   000051        FL.ECH= 000040 G      F.RCTL= 000017        I$$RDN= 000000
CL.CMT= 000001          D$LID   000020        FL.FDX= 000100 G      F.RSIZ= 000002        KDSAR0= 172360
CL.EIF= 000040          D$LNAM  000006        FL.HDX= 000200 G      F.RTYP= 000000        KDSDR0= 172320
CL.EXN= 040000          D$LNUM  000014        FL.KMX= 000400 G      F.SEQN= 000100        KINAR0= 172340
CL.EXT= 000010          D$LST   000047        FL.LLC= 000004 G      F.SPDV= 000072        KINAR5= 172352
CL.IND= 000002          D$MAXC  000064        FL.LMC= 001000 G      F.SPUN= 000074        KINAR6= 172354
CL.LCC= 000020          D$MAXH  000066        FL.MUX= 002000 G      F.STBK= 000036        KINAR7= 172356
CL.NPR= 000400          D$MAXV  000070        FL.PX1= 010000 G      F.UNIT= 000136        KISAR0= 172340
CL.OPN= 000100 G        D$MLL   000040        FL.PX2= 020000 G      F.URBD= 000020        KISAR5= 172352
CL.RNE= 000004          D$MNOD  000041        FL.TOP= 040000 G      F.VBN = 000064        KISAR6= 172354
CL.XIT= 100000          D$NA    000062        FL.WRN= 100000 G      F.VBSZ= 000060        KISAR7= 172356
CMODE = 140000          D$NBEA  000056        FNERR   001364R       F2.ACN= 000020        KISDR0= 172300
CM$CLE= 000002 G        D$NBRA  000054        FNPTR   001362R       F2.AHR= 010000        KISDR6= 172314
CM$SET= 000001 G        D$NEND= 000054        FX.DIR= 000004        F2.DAS= 000001        KISDR7= 172316
CM$SHO= 000004 G        D$NLN   000030        FX.DLN= 000003        F2.DPR= 000400        K$$CNT= 177546
CS.DIF= 000002          D$NN    000060        FX.DMX= 000002        F2.EVT= 000010        K$$CSR= 177546
CS.DVF= 000004          D$OUTT  000043        FX.LEN= 000000        F2.GGF= 002000        K$$LDC= 000000
CS.EQU= 000040          D$RETF  000050        F$$LVL= 000001        F2.IRR= 001000        K$$TPS= 000074
CS.INP= 000001          D$RNN   000002        F.ACTL= 000076        F2.LIB= 000002        LD$LP = 000000
CS.MOR= 000020          D$RTMR  000076        F.ALOC= 000040        F2.MP = 000004        LD.ALI= 000030
CS.NMF= 000001          D$SEG   000036        F.BBFS= 000062        F2.POL= 000100        LD.CEX= 000024
CS.OUT= 000002          D$SER   000032        F.BDB = 000070        F2.RAS= 004000        LD.CIR= 000035
CS.WLD= 000010          D$SQRL  000052        F.BGBC= 000057        F2.RBN= 020000        LD.LIN= 000025
C$$CKP= 000000          D$$BUG= 177514        F.BKDN= 000026        F2.SDW= 000040        LD.LOG= 000021
C$$ORE= 000400          D$$ISK= 000000        F.BKDS= 000020        F2.STP= 100000        LD.MOD= 000037
C$$RSH= 177564          D$$L11= 000001        F.BKEF= 000050        F2.SWP= 040000        LD.NOD= 000027
C.CMLD= 000002          D$$YNC= 000000        F.BKP1= 000051        F2.WND= 000200        LD.OBJ= 000032
C.DEVD= 000006          D$$YNM= 000000        F.BKST= 000024        F3.AST= 000200        LD.PRO= 000026
C.DIRD= 000012          EF    = 000001        F.BKVB= 000064        F3.CLI= 001000        LEVL1 = 000001 G
C.DSDS= 000006          ERRV1   005316R       F.CHR = 000075        F3.CRA= 000001        LF.ACT= 100000 G
C.FILD= 000016          ERRV2   005406R       F.CNTG= 000034        F3.EIS= 000004        LF.BRO= 000400 G
C.MKW1= 000024          ERRV3   005446R       F.DFNB= 000046        F3.NWK= 000002        LF.BWT= 000007 G
C.MKW2= 000026          ERRV4   005616R       F.DSPT= 000044        F3.PMN= 004000        LF.ENA= 002000 G
C.SIZE= 000054          E$$XPR= 000000        F.DVNM= 000134        F3.PRO= 000040        LF.LPB= 001000 G
C.STAT= 000001          FATAL = 176031 G      F.EFBK= 000010        F3.RLK= 020000        LF.MDC= 000100 G
C.SWAD= 000022          FE.CAL= 000040        F.EFN = 000050        F3.SHF= 040000        LF.MFL= 004000 G
C.TYPR= 000000          FE.CEX= 020000        F.EOBB= 000032        F3.STM= 000010        LF.MTP= 000020 G
DEFFLG= 000233          FE.DRV= 000010        F.ERR = 000052        F3.TCM= 002000        LF.PAC= 000200 G
DFCNT   001032RG        FE.DYM= 010000        F.EXT = 000072        F3.UDS= 000020        LF.RDY= 040000 G
DFEXT   001606R         FE.EXP= 000200        F.FACC= 000043        F3.WAT= 010000        LF.REA= 010000 G
DFLTNB  000426R     003 FE.EXT= 000001        F.FFBY= 000014        F3.XHR= 000100        LF.SER= 000040 G
DFPTR   001030RG        FE.EXV= 000004        F.FLG = 000074        F3.11S= 000400        LF.TIM= 000010 G
DFTLEN= 000007          FE.FDT= 002000        F.FNAM= 000110        GCLBUF  000030RG      LF.UNL= 020000 G
DSIZE = 000044 G        FE.LSI= 000400        F.FNB = 000102        GCLSIZ= 000226        LF.X2P= 000000 G
D$AMXC  000072          FE.MUP= 000002        F.FTYP= 000116        G$$TPP= 000000        LINBUF  001000RG      004
```

```
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 10
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

$CLSBK     000176 RG         6-102       #6-102
$CLSEF     000156 RG        #6-102
$CLSIZ   = 000306 RG        #6-102
$CLSRV     000470 RG        #6-102
$CLTBF     000202 RG        #6-102
$CLTSZ     000204 RG        #6-102
$CLWEF     000152 RG        #6-102
$CMAND     002006 RG        #15-457
$CMFRK     003752 RG        #17-785
$CMPDV     003744 RG        #17-777
$CPBIA     000474 RG        #6-125
$CRAVL     004252 RG        #17-937
$CTIM      002364 RG        #15-523
$CUR.N     001260 RG        #11-341
$CXOPT     004070 RG        #17-828
$C5TA    = ******   GX       9-282
$DDFNC     004036 RG        #17-815
$DDMPV     002420 RG        #15-540
$DECPT     004030 RG        #17-812
$DEVHD     003672 RG        #16-750
$DFN       001160 RG        #9-273
$DFNAM     001552 RG         8-242        9-278       #13-392
$DFN2      001162 RG        #9-275
$DFUIC     001534 RG         8-236       #13-391
$DLCPV     002416 RG        #15-539
$ERMSG   = ******   GX       14-436
$EROUT     001674 RG        #14-423       19-1135     19-1137     19-1138     22-1272
$EVDSC     004132 RG        #17-853
$EXSIZ     003674 RG        #16-751
$FEATR     002020 RG        #15-498
$FILHD     004126 RG        #17-850
$FLAGS     003406 RG        #15-567
$FMASK     003676 RG        #16-752       18-1027
$FNOUT     001642 RG        #14-413
$FPDV    = ******   GX       25-1348
$FUNC      002002 RG        #15-455
$GCLEP   = ******   GX       22-1254
$GENND     000720 RG        #6-145
$GENST     000716 RG        #6-144
$GETAD     004712 RG        #21-1191
$GETRC     004432 RG        #18-1033
$GETRD     004432 RG        #18-1034
$GETRV     004366 RG        #18-1001
$HMBUF     000612 RG        #6-140
$ICBI    = 000016  G        #15-467       15-468
$ICBLN     002016 RG        #15-468
$KRBLN   = 000004  G        #15-464
$LDBAF     003736 RG        #17-774
$LGCON     004204 RG        #17-865
$LGDDB     004124 RG        #17-849
$LGFNB     004152 RG        #17-855
$LGMON     004200 RG        #17-864
```

```
                                        .SBTTL  LOCAL DATA

                                ;
                                ; NULL PARAMETER BLOCK
                                ;
000234  000000  000000  000000  NULL:   .WORD   0,0,0,0

                                ;
                                ; UNIT NUMBER
                                ;
000244                          UNIT:   .BLKW

                                ;
                                ; DDM PDV INDEX (SET LINE)
                                ;
000246                          PDVX:   .BLKW   1
                                ;
                                ; STACK POINTER SAVE
                                ;
000250                          SPSAV:  .BLKW   1
                                ;
                                ; UCB BUFFER
                                ;
000252                          UCBBUF: .BLKW   30.             ; LEAVE PLENTY OF ROOM
                                ;
                                ; REQUEST BLOCK TO $FLAGS BIT MAPPING TABLE
                                ;
000346  004000  000000G         LBITS:  .WORD   LS.FDX,FL.FDX   ; SET LINE - FDX, HDX, ECHO, TOP
000352  010000  000000G                 .WORD   LS.HDX,FL.HDX
000356  001000  000000G         PBITS:  .WORD   LS.ECH,FL.ECH   ; SET PROCESS - ECHO, TOP
000362  002000  000000G                 .WORD   LS.TOP,FL.TOP
000366  000000                          .WORD   0
```

```
                                        .SBTTL  SLTA - GET SLT ADDRESS
                                ;+
                                ; SLTA - GET SLT ADDRESS
                                ;
                                ; INPUTS:
                                ;       R3=REQUEST BLOCK POINTER
                                ;               .BLKW   1         ; LR.LIN - LINE NAME
                                ;               .BLKB   1         ; LR.CTL - CONTROLLER NUMBER
                                ;               .BLKB   1         ; LR.UNT - UNIT NUMBER
                                ;               .BLKB   1         ; LR.TRI - TRIBUTARY NUMBER
                                ;
                                ; OUTPUTS:
                                ;       C-BIT=SUCCESS/FAILURE
                                ;       $SLTA=SLT ADDRESS
                                ;       $SLN=SYSTEM LINE NUMBER
                                ;-
000774  012300                  SLTA:   MOV     (R3)+,R0          ; GET LINE NAME
000776  001516                          BEQ     101$              ; IF ZERO, RETURN FAILURE
001000  010067  000000G                 MOV     R0,$NAME          ; SAVE IT
001004                                  CALL    $FPDV             ; LOOK FOR PDV ADDRESS
001010  103511                          BCS     101$              ; IF CS, NO SUCH PROCESS
001012  132760  000000G 000000G         BITB    #ZF.DDM,Z.FLG(R0) ; IS DDM BIT SET IN PDV?
001020  001505                          BEQ     101$              ; IF EQ, NO
001022  010067  000000G                 MOV     R0,$DDMPV         ; SAVE DDM PDV ADDRESS
001026  010167  000246'                 MOV     R1,PDVX           ; AND PDV INDEX
001032  132760  000000G 000000G         BITB    #ZF.LMC,Z.FLG(R0) ; MICRO-CODE LOAD NEEDED?
001040  001403                          BEQ     5$                ; IF EQ, NO
001042  052767  000000G 000000G         BIS     #FL.LMC,$FLAGS    ; UPDATE NETLDR STATUS
001050  005046                  5$:     CLR     -(SP)             ; SET MAXIMUM UNIT COUNT FOR NON-MUX
001052  132760  000000G 000000G         BITB    #ZF.MUX,Z.FLG(R0) ; MUX DEVICE?
001060  001416                          BEQ     10$               ; IF EQ, NO
001062  052767  000000G 000000G         BIS     #FL.MUX,$FLAGS    ; UPDATE NETLDR STATUS
001070  012716  000017                  MOV     #15.,(SP)         ; MAXIMUM OF 16. UNITS/CONTROLLER
001074  132760  000000G 000000G         BITB    #ZF.KMX,Z.FLG(R0) ; COMIOP MUX DEVICE?
001102  001405                          BEQ     10$               ; IF EQ, NO
001104  052767  000000G 000000G         BIS     #FL.KMX,$FLAGS    ; UPDATE NETLDR STATUS
001112  012716  000057                  MOV     #47.,(SP)         ; MAXIMUM OF 48. UNITS/CONTROLLER
001116  005001                  10$:    CLR     R1                ; GET CONTROLLER NUMBER
001120  152301                          BISB    (R3)+,R1          ;
001122  005002                          CLR     R2                ; AND UNIT NUMBER
001124  152302                          BISB    (R3)+,R2          ;
001126  022602                          CMP     (SP)+,R2          ; VALID UNIT NUMBER?
001130  103444                          BLO     111$              ; IF LO, NO
001132  112367  000000G                 MOVB    (R3)+,$TRIB       ; SAVE TRIBUTARY NUMBER
001136  105723                          TSTB    (R3)+             ; SKIP NEXT ENTRY
001140                                  CALL    $FSLT             ; TRY TO FIND SLT
001144  103433                          BCS     101$              ; IF CS, NOT FOUND
001146  010067  000000G                 MOV     R0,$SLTA          ; SET SLT ADDRESS
001152  010167  000000G                 MOV     R1,$SLN           ; AND SYSTEM LINE NUMBER
001156  010267  000000G                 MOV     R2,$$MUX          ; SAVE UNIT COUNT (FOR $NLCFG)
001162  126760  000000G 000000G         CMPB    $TRIB,L.NSTA(R0)  ; IS THE TRIBUTARY NUMBER VALID ?
001170  101032                          BHI     131$              ; IF HI, NO
001172  116001  000000G                 MOVB    L.DLC(R0),R1      ; GET DLC PDV INDEX
001176  066701  000000C                 ADD     $PDVTA+AB,R1      ; ADD PDV TABLE ADDRESS
001202  016102  000000G                 MOV     AB(R1),R2         ; GET DLC PDV ADDRESS

001206  132762  000000G 000000G 20$:    BITB    #ZF.DLC,Z.FLG(R2) ; IS DLC BIT SET IN PDV?
```

```
                                        .SBTTL  $BITS - COPY BITS TO $FLAGS
                                ;+
                                ; $BITS - COPY BITS FROM REQUEST BLOCK TO $FLAGS
                                ;
                                ; INPUTS:
                                ;       R0 = POINTER INTO MAPPING TABLE
                                ;
                                ; OUTPUTS:
                                ;       R1 = ADDRESS OF REQUEST BLOCK STATUS WORD
                                ;       R2 = ADDRESS OF $FLAGS
                                ;
                                ;-
002744                          $BITS:
002744  012701  000004G                 MOV     #$$LRB+4,R1     ; SETUP ADDRESSES
002750  012702  000000G                 MOV     #$FLAGS,R2      ;
002754  000401                          BR      20$             ; GO TEST FIRST ITEM
002756  005720                  10$:    TST     (R0)+           ; SKIP SECOND WORD IN TABLE ENTRY
002760  005710                  20$:    TST     (R0)            ; END OF TABLE ?
002762  001404                          BEQ     30$             ; IF EQ, YES
002764  032011                          BIT     (R0)+,(R1)      ; IS THE BIT SET IN REQUEST BLOCK ?
002766  001773                          BEQ     10$             ; IF EQ, NO
002770  052012                          BIS     (R0)+,(R2)      ; SET THE BIT IN $FLAGS
002772  000772                          BR      20$
002774                          30$:    RETURN
```

```
                                                                  I  1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 5
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE             REFERENCES

FF.QUL  = 000370          #5-77
FF.REM  = 100000          #5-77
FIELDC  = 000003          #7-194     98-4280
FILBUF    000222 RG       #7-176     7-177     90-3819   90-3846   90-3847  *90-3851  *90-3852   101-4406  101-4407
                           103-4551
FILES     014142 RG       #90-3819
FILFLG    000252 RG       #7-181
FINFIL    013104 R         79-3512  #80-3534
FINOBJ    013632 R         86-3693  #87-3718
FINREM    013732 R         69-3181  #88-3745
FLAG      000062 R        #6-129    *84-3629  *85-3647   86-3682
FLSZ    = 000020  G       #7-177     103-4552
FLTFLG    000044 R        #6-124    *77-3443  *79-3487  *79-3489  *79-3491   79-3503
FNBEXT  = 000006          #7-193     101-4410
FNBLN     000264 RG       #7-187    *101-4411  102-4432  102-4443  102-4451 *103-4541
FNBSZ   = 000044  G       #7-190     98-4272   101-4389
FNBUFF    000266 RG       #7-189     7-190     7-191     7-192     90-3826   90-3857   98-4276   101-4388  102-4444
                           103-4539  103-4599  104-4630  104-4676
FPDV      015402 R         92-3938  #96-4179
FSLT      015450 R         92-3946  #97-4211
F.ADD     000016          #5-77
F.CEV     000030          #5-77
F.CIR     000024          #5-77
F.CLS     000002          #5-77     *79-3497  *79-3498   80-3543
F.EVT     000004          #5-77     *79-3499  *79-3500  *79-3501  *79-3502
F.FLG     000014          #5-77     *79-3503
F.HIBK  = ******    GX     17-1087
F.L[illegible]     000016          #5-77      79-3493   79-3498   79-3506   80-3542
F.LIN     000020          #5-77
F.LNK     000000          #5-77
F.MOD     000022          #5-77
F.NRBD  = ******    GX    *8-292    *8-294    *8-373    *11-741   *11-743   *61-2777  *64-2892  *64-2895  *65-2923
                          *65-2959  *65-2961  *69-3190  *69-3192  *94-4075  *94-4077  *102-4466 *102-4468
F.NXT     000000          #5-80
F.REM     000026          #5-77
F.RSIZ  = ******    GX    *8-329    *8-402    *8-418    *8-448    *8-457    *11-742   *12-797   *13-834   *14-879
                          *16-948   *26-1569  *61-2753  *61-2778  *62-2792  *62-2808  *63-2841  *63-2863  *64-2889
                          *65-2920  *65-2926  *65-2956  *65-2960  *67-3017  *67-3027  *67-3037  *67-3046  *67-3056
                          *67-3065  *67-3074  *67-3083  *67-3093  *67-3103  *67-3110  *69-3175  *69-3184  *69-3191
                          *70-3239  *71-3257  *72-3280  *73-3303  *74-3318  *75-3344  *75-3372  *76-3392  *76-3408
                          *79-3506  *80-3542  *86-3686  *87-3726  *88-3747  *88-3762  *94-4060  *94-4063  *94-4076
                          *98-4258  *98-4272  *102-4451 *102-4454 *102-4467 *103-4524 *103-4526 *103-4530 *103-4536
                          *105-4707
F.SEV     000004          #5-77
F.URBD  = ******    GX    *8-286    *8-288    *61-2752  *61-2754  *64-2888  *64-2890  *65-2919  *65-2922  *69-3183
                          *69-3185  *88-3746  *88-3748  *94-4062  *94-4064  *98-4257  *98-4259  *102-4453 *102-4455
F.USE     000002          #5-80      98-4277   98-4278   103-4524 *103-4525  103-4526
F.VAR     000003          #5-80
GETSTR    016760 RG       #103-4569
GRPUIC    017062 RG       #104-4618
HEXIND    015330 R         93-3991   93-3999   95-4115   95-4117  #96-4138
HEXSTR    000164 RG       #7-170     95-4097   95-4102   95-4113

                                                                  J  1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 6
```

```
                                                                 I  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 18
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE            REFERENCES

                          102-4467
$DEUMR   = ******  GX     16-1018
$DFUIC   = ******  GX     29-1726
$DIGIT   = 000024        #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$DNUMB   = 000014        #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$EOS     = 000012        #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$EXIT    = 000000        #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$FAIL    = 177777        #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$FILHD   = ******  GX     79-3513     80-3534
$FMASK   = ******  GX     8-225       8-327       16-939
$GCLEP   = ******  GX     28-1702     30-1752
$GETAD   = ******  GX     65-2926     80-3542     87-3726     88-3762     98-4272     103-4524    103-4530    105-4707
$GETRV   = ******  GX     8-270       8-287       8-329       8-457       16-948      17-1054     26-1573     26-1577     26-1583
                          26-1594     26-1597     26-1607     26-1614     61-2753     62-2792     62-2808     63-2841     63-2863
                          64-2889     65-2920     67-3017     67-3027     67-3037     67-3046     67-3056     67-3065     67-3074
                          67-3083     67-3093     67-3103     67-3110     69-3184     70-3239     71-3257     72-3280     73-3303
                          74-3318     75-3344     75-3372     76-3392     76-3408     88-3747     94-4063     98-4258     102-4454
$GPRM    = ******         18-1108     19-1139     20-1178     21-1220     22-1250     23-1265     32-1884     46-2329
$GTBUF     003132 RG      13-839      14-884     #15-908
$HMBUF   = ******  GX     61-2752     61-2758     64-2888     64-2892    *64-2893     65-2919     65-2921     65-2954     65-2959
                          69-3183     69-3187     69-3190     88-3746     88-3750     94-4062     94-4066     94-4075     98-4257
                          98-4261     102-4453    102-4457    102-4466
$LAMDA   = 000000        #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$LGSTT   = ******  GX    *81-3563
$LSTAD   = ******  GX    *88-3757     89-3788     89-3793    *89-3795
$LSTRM   = ******  GX    *88-3758     89-3796    *89-3797
$MAC     = ******  GX     29-1728
$MAXOV   = ******  GX     13-828      14-873      53-2592    *54-2610    *54-2611
$MFL       017360 RG      8-511      #106-4726
$MUL     = ******  GX     24-1355     24-1364     24-1388     24-1417     24-1430     24-1443
$NBIAS   = ******  GX    *8-408      *8-414
$NTCFP   = ******  GX     8-460       8-462
$NTCFS   = ******  GX     8-465       8-467
$NTLHB   = ******  GX    *8-374      *8-375      *8-476      *8-486
$NUMBR   = 000002        #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$OBJHD   = ******  GX     86-3694     87-3719
$OPTON   = ******  GX    *107-4778   *108-4816   *109-4850
$PARHD   = ******  GX     8-266       17-1042
$PAVL    = ******  GX    *8-378
$PDVNM   = ******  GX     28-1686     56-2639     96-4181     106-4731
$PDVTA   = ******  GX     28-1685     56-2638     59-2693     96-4180     96-4192     106-4730    106-4749
$PUMR    = ******  GX    *8-427
$PUTAD   = ******  GX     8-402       8-418       8-448       13-834      14-879      65-2956     69-3175     79-3506     86-3686
                          94-4060     102-4451    103-4526    103-4536    105-4709
$PUTRC   = ******  GX     8-293       8-298       8-337       8-459       11-742      12-797      26-1569     26-1587     26-1592
                          26-1611     26-1616     61-2778     62-2801     62-2821     63-2844     63-2868     64-2894     65-2960
                          67-3020     67-3030     67-3039     67-3048     67-3058     67-3067     67-3076     67-3085     67-3095
                          67-3105     67-3112     69-3191     70-3243     71-3266     72-3291     73-3306     74-3330     75-3357
                          75-3376     76-3396     76-3420     94-4076     102-4467
$QBIAS   = ******  GX    *8-443
$QSTRT   = ******  GX    *8-444
$QUAL    = ******  GX    *107-4780   *108-4818   *109-4852
```

```
    121                                          .SBTTL  LOCAL DATA
    122                                 ;***
    123                                 ; LOCAL DATA
    124                                 ;***
    125
    126 000000                                   .PSECT  DATA,D
    127
    128
    129                                 ;
    130                                 ; SPECIAL CASE ERROR MESSAGES
    131                                 ;
    132
    133 000000                          ERMSG$  <%N VNP -- Microcode File Open Failure%N>
    134 000047                          ERBLK$  UNAOP,MSGPRI
    135
    136 000054                          ERMSG$  <%N VNP -- Microcode File Label Block Read Error%n>
    137 000135                          ERBLK$  UNARE,MSGPRI
    138
    139                                 ;
    140                                 ; PROCESS SIZES (INCLUDING EXTENSION)
    141                                 ;
    142 000142                          BLOX:   .BLKW   1               ; 32. WORD BLOCKS
    143 000144                          BYTES:  .BLKW   1               ; BYTES
    144
    145                                 ;
    146                                 ; PROCESS LINE TABLE/CONTROLLER TABLE EXTENSION SIZE
    147                                 ;
    148 000146                          EXTEND: .BLKW   1
    149
    150                                 ;
    151                                 ; MICROCODE LOAD PARTITION DATA
    152                                 ;
    153         000001                  MCLOA   = 1
    154 000150  000000                  MCFLG:  .WORD   0
    155 000152                          TMPPCB: .BLKW   1
    156 000154     125     116     101  UNAMC:  .ASCIZ  /UNAMC.TSK/
        000157     115     103     056
        000162     124     123     113
        000165     000
    157 000166     113     104     120  KDPMC:  .ASCIZ  /KDPMC.TSK/
        000171     115     103     056
        000174     124     123     113
        000177     000
    158 000200     113     104     132  KDZMC:  .ASCIZ  /KDZMC.TSK/
        000203     115     103     056
        000206     124     123     113
        000211     000
    159
    160                                 .EVEN
    161
    162 000212                          MCFBF:  .BLKB   30.
    163
    164                                 ;
    165                                 ; ERROR INDICATION
    166                                 ;
    167 000250                          ERROR:  .BLKW   1
    168
```

```
001740  066701  000016G                  ADD     $BFR+P.SIZE,R1   ; GET TOP OF PARTITION

                                         .IF NDF R$$MPL                                               ;[MP01]
001744  032767  000000G 000000G          BIT     #VF.SS,$VFLAG    ; 11S SYSTEM ?
001752  001415                           BEQ     45$              ; IF NO - BRANCH
                                         .ENDC                                                        ;[MP01]

001754  012702  000000G                  MOV     #SYSFDB,R2       ; GET FDB ADDRESS
001760  016202  000002G                  MOV     F.HIBK+2(R2),R2  ; GET MAX VBN
001764  162702  000002                   SUB     #2,R2            ; CALCULATE TOP OF MEMORY IN IMAGE
001770  000241                           CLC                      ; ...
001772  006302                           ASL     R2               ; ...
001774  006302                           ASL     R2               ; ...
001776  006302                           ASL     R2               ; ...
002000  020201                           CMP     R2,R1            ; LEGAL BIAS ?
002002  103007                           BHIS    50$              ; IF YES - BRANCH
002004  000403                           BR      47$              ; CONTINUE

002006  022701  007600           45$:    CMP     #7600,R1         ; WITHIN 124KW (M ONLY)
002012  103003                           BHIS    50$              ; IF YES - BRANCH
002014  042767  000000G 000000G  47$:    BIC     #FL.TOP,$FLAGS   ; IF NO - LOAD FIRSTFIT

002022                           50$:    CALL    PR.SUB           ; CREATE SUB-PARTITION
002026  000503                           BR      80$              ;
002030  032767  000000G 000000G  60$:    BIT     #FL.CEX,$FLAGS   ; IS THIS THE COMM EXEC?
002036  001430                           BEQ     70$              ; IF EQ, NO
002040  016701  000014G                  MOV     P.REL+$BFR,R1    ; GET PARTITION START IN BLOCKS
002044  010102                           MOV     R1,R2            ; COPY IT
002046                                   ASL$    6,R1             ; CONVERT TO BYTES
002062  026001  000000G                  CMP     L$BSA(R0),R1     ; WAS CEX BUILT FOR THIS PARTITION?
002066  001076                           BNE     130$             ; IF NE, NO
002070  066702  000016G                  ADD     P.BLKS+$BFR,R2   ; ADD SIZE TO START ADDRESS
002074  022702  001000                   CMP     #1000,R2         ; IS PARTITION WITHIN FIRST 16K?
002100  103007                           BHIS    70$              ; IF HIS, YES
002102  032767  000004  000000G          BIT     #FE.EXV,$FMASK   ; IS 20K EXEC SUPPORTED?
002110  001505                           BEQ     170$             ; IF EQ, NO
002112  022702  001200                   CMP     #1200,R2         ; IS PARTITION WITHIN FIRST 20K?
002116  103502                           BLO     170$             ; IF LO, NO
002120  026767  000142' 000016G  70$:    CMP     BLOX,P.BLKS+$BFR ; IS PARTITION BIG ENOUGH ?
002126  101062                           BHI     140$             ; IF HI, NO

002130  032767  000200  000030G          BIT     #PS.COM,P.STAT+$BFR ; COMMON PARTITION?
002136  001462                           BEQ     150$             ; IF EQ, NO
002140  105767  000025G                  TSTB    P.BUSY+1+$BFR    ; IS PARTITION BUSY?
002144  001063                           BNE     160$             ; IF NE, YES
002146  016704  000012G                  MOV     P.MAIN+$BFR,R4   ; GET MAIN PARTITION PCB ADDRESS
002152                                   GETRV   R4               ; READ IN PCB
002162  105767  000025G                  TSTB    P.BUSY+1+$BFR    ; IS MAIN PARTITION BUSY?
002166  001052                           BNE     160$             ; IF NE, YES
002170                                   GETRV   R5               ; READ IN PCB
002200  156767  000024G 000025G          BISB    P.BUSY+$BFR,P.BUSY+1+$BFR ; SET PARTITION BUSY
002206                                   PUTRC                    ; REWRITE PCB
002212  016746  000024G                  MOV     P.BUSY+$BFR,-(SP) ; SAVE BUSY FLAG
002216                                   GETRV   R4               ; READ IN PCB
002226  152667  000025G                  BISB    (SP)+,P.BUSY+1+$BFR ; SET MAIN PARTITION BUSY
002232                                   PUTRC                    ; REWRITE PCB

```

```
VPIO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:04     PAGE 5
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME         REFERENCES

ASL$           #5-58     #5-82      11-460    12-549    12-550    14-701
ASR$           #5-58      12-542    13-610
CACDF$         #5-59      5-71
CALL            9-268     9-273     9-275     9-277     9-278     9-283     9-292     9-295     9-298     10-333
                10-351    10-352    10-355    10-358    10-361    10-363    10-372    10-374    10-383    10-386
                10-394    10-396    10-399    11-455    12-489    12-517    12-530    12-545    13-599    13-606
                14-650    14-695    14-719    14-722    14-724    14-726    14-728    15-773    15-805    15-809
                15-813    15-819    15-822    15-826    15-828    15-831    15-851    15-854    15-857    15-859
                16-901    16-904    17-960    17-962    17-963
CALLR           15-870
CRASH           9-310
CTRDF$         #5-59      5-70
DHBDF$         #5-58      5-69
ECDDB$         #5-59      5-72
ERBLK$         #5-59      7-134     7-137
ERMSG$         #5-59      7-133     7-136
ERRPT$         #5-59      10-394    10-399
GETAD          #5-58
GETRV          #5-58      9-292     10-333    10-383    13-599    14-650    14-719    14-722    14-726    15-809
                15-813    15-819    15-828    15-831    15-851    15-857    16-901
HWDDF$         #5-58      5-67
PCBDF$         #5-58      5-68
PUTAD          #5-58
PUTRC          #5-58      9-295     10-386    14-724    14-728    15-805    15-822    15-826    15-854    15-859
RAND           #9-292     9-292    #9-295     9-295    #10-333    10-333   #10-383    10-383   #10-386    10-386
               #13-599    13-599   #14-650    14-650   #14-719    14-719   #14-722    14-722   #14-724    14-724
               #14-726    14-726   #14-728    14-728   #15-805    15-805   #15-809    15-809   #15-813    15-813
               #15-819    15-819   #15-822    15-822   #15-826    15-826   #15-828    15-828   #15-831    15-831
               #15-851    15-851   #15-854    15-854   #15-857    15-857   #15-859    15-859   #16-901    16-901
RETURN          9-301     10-379    10-391    11-441    11-464    12-562    12-578    13-614    14-736    15-861
                15-867    16-906    16-913    17-947    17-965
SWTCHI         #5-58
```

FILEID**VPRMPT

```
                .TITLE  VPRO - VNP SET PROCESS ROUTINES
                .IDENT  /V05.00/

        ;
        ; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
        ;
        ;
        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
        ; TRANSFERRED.
        ;
        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
        ; CORPORATION.
        ;
        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
        ;
        ; MODULE DESCRIPTION:
        ;
        ;       VNP - "SET PROCESS" ROUTINES
        ;
        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
        ;
        ; IDENT HISTORY:
        ;
        ; 1.00  27-FEB-78
        ;       VERSION 2.0 RELEASE
        ;
        ; 2.00  14-DEC-79
        ;       DECNET-11M/S V3.0
        ;       DECNET-11M-PLUS V1.0
        ;
        ; 3.00  16-APR-82
        ;       DECNET-11M V3.1
        ;       DECNET-11M-PLUS V1.1
        ;
        ; 4.00  07-NOV-83
        ;       DECNET-11M V4.0
        ;       DECNET-11M-PLUS V2.0
        ;
        ; 5.00  22-JUL-85
        ;       DECnet-11M/S V4.2
        ;       DECnet-11M-Plus V3.0
        ;       DECnet-Micro/RSX V1.0
        ;
        ;
```

```
                                                .SBTTL  $ROL - ALLOCATE REACHABILITY TABLES
                                        ;+
                                        ; $ROL - ALLOCATE REACHABILITY TABLES
                                        ;
                                        ; INPUTS:
                                        ;               $PDVA - XPT PDV ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;               REACHABILITY TABLES ALLOCATED
                                        ;
                                        ;               N$ROA1   - NUMBER OF ENTRIES IN NODE REACHABILITY TABLE
                                        ;               N$ROA1+2 - BIAS OF NODE REACHABILITY TABLE
                                        ;               N$ROA1+4 - VIRTUAL ADDRESS OF NODE REACHABILITY TABLE
                                        ;               N$ROA2   - NUMBER OF ENTRIES IN AREA REACHABILITY TABLE
                                        ;               N$ROA2+2 - BIAS OF AREA REACHABILITY TABLE
                                        ;               N$ROA2+4 - VIRTUAL ADDRESS OF AREA REACHABILITY TABLE
                                        ;-

001452                                  $ROL::  GETRV   $DECPT,#D$END           ; READ DECNET HOME BLOCK
001472  005767  000060G                         TST     $BFR+D$NN               ; GET MAX NODES
001476  001534                                  BEQ     70$                     ; IF ENDNODE - BRANCH

001500  016704  000060G                         MOV     $BFR+D$NN,R4            ; STORE MAX NODES
001504  010401                                  MOV     R4,R1                   ; COPY MAX NODES
001506  016705  000062G                         MOV     $BFR+D$NA,R5            ; STORE MAX AREAS
001512  001401                                  BEQ     10$                     ; IF NONE - BRANCH
001514  005201                                  INC     R1                      ; INCLUDE EXTRA AREA ENTRY
001516  005201                          10$:    INC     R1                      ; INCLUDE EXTRA NODE ENTRY
001520  060501                                  ADD     R5,R1                   ; INCLUDE AREAS

001522  006301                                  ASL     R1                      ; CONVERT TO BYTES
001524  062701  000077                          ADD     #77,R1                  ; ROUND UP TO NEXT BLOCK
001530  042701  000077                          BIC     #77,R1                  ; ...
001534                                          ASR$    6,R1                    ; CALCULATE NUMBER OF BLOCKS

001550                                          CALL    $ALN18                  ; ALLOCATE POOL..
001554  103506                                  BCS     80$                     ; THIS SHOULD NEVER OCCUR, BUT

001556  016702  000000G                         MOV     $PDVA,R2                ; GET XPT PDV ADDRESS
001562  016202  000000G                         MOV     Z.DAT(R2),R2            ; GET XPT DDB ADDRESS
001566                                          GETRV   R2,#N$XLEN              ; READ XPT DDB
001604  010067  000024G                         MOV     R0,N$ROA1+2+$BFR        ; STORE BIAS OF NODE ROL
001610  010403                                  MOV     R4,R3                   ; GET MAX NODES

001612  005203                                  INC     R3                      ; INCREASE BY ONE
001614  010367  000022G                         MOV     R3,N$ROA1+$BFR          ; STORE NUMBER OF ENTIRES IN NODE ROL
001620  005002                                  CLR     R2                      ; SET VIRTUAL ADDRESS
001622                                          BIAS    R2                      ; ...
001632  010267  000026G                         MOV     R2,N$ROA1+4+$BFR        ; ...

001636  010501                                  MOV     R5,R1                   ; GET MAX AREAS
001640  001413                                  BEQ     20$                     ; IF LEVEL1 - BRANCH

001642  005201                                  INC     R1                      ; INCREASE BY ONE
001644  010167  000030G                         MOV     R1,N$ROA2+$BFR          ; STORE NUMBER OF ENTRIES IN AREA ROL
001650  016767  000024G 000032G                 MOV     N$ROA1+2+$BFR,N$ROA2+2+$BFR ; STORE BIAS OF AREA ROL
001656  006303                                  ASL     R3
```

```

                        .SBTTL  $CTSYM - FIND SYMBOLS FROM CETAB                          ;[MP01]
                ;+
                ; *** - $CTSYM - FIND SYMBOLS FROM CETAB
                ;
                ; INPUT:
                ;       $NAME = RAD50 PROCESS NAME
                ;
                ; OUTPUT:
                ;       C-BIT = SUCCESS/FAILURE
                ;       $CTBAD = CTB ADDRESS FOR THIS PROCESS (ZERO IF NOT A DDM)
                ;
                ;-
004504          $CTSYM::
                        .IF DF  R$$MPL
                        MOV     $NAME,CTBSYM+2  ; FORM SYMBOL NAME TO LOOK FOR "$CTXXX"
                        MOV     #$DFUIC,R0      ; GET THE COMM EXEC UIC
                        MOV     #CETNAM,R1      ; FORM ASCIZ FILE SPEC "[XX,5X]CETAB.STB"
                        MOV     #STBEXT,R2      ;       ...
                        CALL    $AZFS           ;       ...
                        MOV     #CTBLK,R0       ; GET SYMBOL BLOCK TO LOOK FOR
                        CALL    $SYMBL          ; LOOK FOR IT
                        TST     CTBLK+2         ; WAS SYMBOL DEFINED ?
                        BEQ     20$             ; IF EQ, NO .. ASSUME IT IS NOT A DDM

                        SWTCHI                  ; SET DEFAULT BUFFERS
                        SWTCHO                  ;   ...
                        MOV     $NTLHB+.CXALL,R1 ; GET START OF CETAB
                        ADD     CTVAL,R1        ; CALCULATE FIRST CTB ADDRESS
                        MOV     R1,$CTBIM       ; SAVE CTB ADDRESS
                        MOV     R1,$CTBAD       ; SAVE CTB ADDRESS AGAIN


                ;
                ; SEE IF CTB ALREADY IN LIST
                ;

                        MOV     $CTLST,$BFR     ; GET START OF CTB LIST
                3$:     MOV     $BFR,R0         ; STORE NEXT CTB ADDRESS
                        BEQ     5$              ; IF END OF LIST - BRANCH
                        GETRV   R0,#L.KRB+2     ; READ CTB
                        CMP     R0,R1           ; IS THIS OUR CTB ?
                        BNE     3$              ; IF NE, NO .. KEEP LOOKING

                        MOV     #$CETAB,R0      ; CALCUALTE CTB ADDRESS IN VNP TEMPLATE
                        ADD     CTVAL,R0        ;   ...
                        MOV     R0,CTBSAV       ;   ...
                        MOV     R0,$CTBAD       ; SAVE THIS FOR VALT ROUTINE
                        RETURN                  ; CTB ALREADY IN LIST

                5$:
                        MOV     #$CETAB,R0      ; CALCUALTE CTB ADDRESS IN VNP TEMPLATE
                        ADD     CTVAL,R0        ;   ...
                        MOV     R0,CTBSAV       ;   ...
                        MOV     R0,$CTBAD       ; SAVE THIS FOR VALT ROUTINE
                        MOV     $CTLST,L.LNK(R0) ; LINK CTB INTO FRONT OF CTB CHAIN

```

```
**FILE**ID**VRES





VV      VV    RRRRRRRR     EEEEEEEEEE      SSSSSSSS
VV      VV    RRRRRRRR     EEEEEEEEEE      SSSSSSSS
VV      VV    RR      RR   EE            SS
VV      VV    RR      RR   EE            SS
VV      VV    RR      RR   EE            SS
VV      VV    RR      RR   EE            SS
VV      VV    RRRRRRRR     EEEEEEEE        SSSSSS
VV      VV    RRRRRRRR     EEEEEEEE        SSSSSS
VV      VV    RR  RR       EE                    SS
VV      VV    RR  RR       EE                    SS
  VV  VV      RR    RR     EE                    SS        ....
  VV  VV      RR    RR     EE                    SS        ....
    VV        RR      RR   EEEEEEEEEE    SSSSSSSS          ....
    VV        RR      RR   EEEEEEEEEE    SSSSSSSS          ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                        .SBTTL  SLFNP - SETUP LOCAL FILE NAME POINTER
                                ;+
                                ; SLFNP - SETUP LOCAL FILE NAME POINTER
                                ;
                                ; INPUTS:
                                ;       R0=NUMBER TO ADD TO GCL LEVEL INDICATOR (0=CURRENT, 1=PREVIOUS)
                                ;
                                ; OUTPUTS:
                                ;       R0=LOCAL FILE NAME AREA ADDRESS
                                ;-
001332  010046                  SLFNP:  MOV     R0,-(SP)                ; SAVE CURRENT/PREVIOUS INDICATOR
001334  116700  000206'                 MOVB    $CLEVL,R0               ; GET CURRENT GCL LEVEL
001340  062600                          ADD     (SP)+,R0                ; ADJUST LEVEL BY ORIGINAL R0 VALUE
001342  006300                          ASL     R0                      ; 2X
001344  010046                          MOV     R0,-(SP)                ; SAVE 2X VALUE
001346  006300                          ASL     R0                      ; 4X
001350  006300                          ASL     R0                      ; 8X
001352  062600                          ADD     (SP)+,R0                ; 10X
001354  062700  001414'                 ADD     #NAMES-10.,R0           ; OFFSET INTO LOCAL STORAGE AREA
001360                                  RETURN
```

```
                                        .SBTTL  $GETRC,$GETRV - GET A RECORD FORM SYSTEM IMAGE FILE
                                ;+
                                ;       **-$GETRC
                                ;       **-$GETRV
                                ;
                                ;       GET A RECORD FROM THE SYSTEM IMAGE FILE. THE RECORD
                                ;       CANNOT BE MORE THAN 512. BYTES BUT CAN CROSS BLOCK
                                ;       BOUNDARIES. THE ADDRESS SUPPLIED TO THIS ROUTINE IS
                                ;       ASSUMED TO BE IN THE EXECUTIVE VIRTUAL ADDRESS SPACE.
                                ;       THE ADDRESS IS VALIDATED ; I.E. IT CHECKS                         [MP01]
                                ;       TO BE SURE THAT THE ENTIRE RECORD DESCIBED BY $RADDR &
                                ;       $RSIZE IS ENTIRELY WITHIN THE EXECUTIVE ADDRESS
                                ;       SPACE.
                                ;
                                ; INPUTS:
                                ;       $RADDR  -ADDRESS OF THE RECORD.
                                ;       $RSIZE  -SIZE OF THE RECORD.
                                ;
                                ; OUTPUTS:
                                ;       ALL ENTRIES WILL CHECK IF THE RECORD ADDRESS IS INVALID, THEN A [MP01]
                                ;       FATAL ERROR IS DECLARED. ELSE THE SPECIFIED RECORD IS READ FROM
                                ;       THE TASK IMAGE FILE INTO $BFR.
                                ;
                                ; NOTE: ALL REGISTERS ARE PRESERVED ACROSS CALL.
                                ;-

                                        .MCALL  HWDDF$
004366                                  HWDDF$                  ; HARDWARE OFFSETS

                                        .ENABL  LSB
004366                          $GETRV::
                                        .IF DF  R$$MPL                                          ;[MP01]

                                        BIT     #160000,$RADDR  ; ADDRESS WITHIN FIRST 4KW ?    ;[MP01]
                                        BEQ     1$              ; IF YES - BRANCH               ;[MP01]

                                        BIT     #DF.DAS,$DFLAG  ; D-SPACE ENABLED ?             ;[MP01]
                                        BEQ     1$              ; IF NO - BRANCH                ;[MP01]
                                        MOV     $RBLCK,RB       ; SAVE BIAS                     ;[MP01]
                                        MOV     .DBIAS,$RBLCK   ; IF YES - MAP TO RSX POOL      ;[MP01]
                                        MOV     $RADDR,RA       ; SAVE VIRTUAL ADDRESS          ;[MP01]

                                        CALL    DBLK            ; CALCULATE BLOCK AND ADDRESS   ;[MP01]
                                        CALL    $GETAD          ; TAKE A LEAP                   ;[MP01]
                                        MOV     RA,$RADDR       ; RESTORE VIRTUAL ADDRESS       ;[MP01]
                                        MOV     RB,$RBLCK       ; RESTORE BIAS                  ;[MP01]
                                        RETURN
                                1$:                                                             ;[MP01]
                                        .ENDC                                                   ;[MP01]

004366  016746  174064                  MOV     $RADDR,-(SP)    ; GET ADDRESS OF RECORD
004372  032716  000001                  BIT     #1,(SP)         ; ODD ADDRESS?
004376  001122                          BNE     30$             ; IF NE YES
004400  066716  173654                  ADD     $RSIZE,(SP)     ; CALCULATE HIGHEST ADDRESS
004404  103517                          BCS     30$             ; IF CS ADDRESS OVERFLOW
004406  012746  120000                  MOV     #120000,-(SP)   ; ASSUME 20K EXEC
004412  032767  000004  177256          BIT     #FE.EXV,$FMASK  ; 20K EXEC SUPPORTED?
```

```
VRES - VNP ROOT MACRO V05.03b Tuesday 17-Sep-85 16:15  Page 26-2
Symbol table

LINSIZ  001024RG        LO.VCT  000014          L.DDM   000002 G        N.STAT= 000020          OP$STA= 000004
LN.CLO= 000000 G        LO.VER  000042          L.DDS   000004 G        N.UNIT= 000034          OP$TPA= 000020
LN.DUM= 000005 G        LR.ACC  000020          L.DLC   000003 G        OF$INS= 000001          OP$TRI= 000040
LN.LOA= 000004 G        LR.ACO  000052          L.DLM   000006 G        OF$LOG= 000100          OP$UCS= 000020
LN.LOO= 000003 G        LR.ADD  000012          L.DLS   000010 G        OF$OFF= 000002          OP$USE= 000002
LN.OAU= 000003 G        LR.ALI  000002          L.FLG   000000 G        OF$ON = 000000          OP$VEC= 000040
LN.OFF= 000001 G        LR.BLK  000010          L.KRBA  000016 G        OF$SMC= 000200          OP$VER= 000040
LN.ON = 000000 G        LR.CTL  000004          L.LEN = 000022 G        OP$ADD= 000200          OP$XAC= 000001
LN.OOP= 000004 G        LR.DES  000010          L.MPF   000022 G        OP$ALL= 000200          OP$XPR= 000002
LN.OPE= 000001 G        LR.HLP  000002          L.NMST  000020 G        OP$CCS= 000004          OP$X5S= 000004
LN.REF= 000002 G        LR.LIN  000002          L.NSTA  000014 G        OP$CIE= 004000          OP$X9S= 000010
LN.SER= 000002 G        LR.NAM  000012          L.OWNR  000021 G        OP$CON= 000400          PBLOX   001004RG
LN.STA= 000017 G        LR.PAS  000041          L.UNT   000013 G        OP$COP= 000001          PDVNAM  001036RG
LN.SUB= 000360 G        LR.PRO  000002          MACTYP  001034RG        OP$COS= 000100          PF.PSV= 000002 G
LN.TRI= 000006 G        LR.STS  000000          MAXOV   000476RG        OP$DST= 000200          PF.XDF= 000001 G
LO.ADD  000022          LR.TRI  000006          MPAR  = 172100          OP$DTE= 000040          PF.XGA= 000004 G
LO.ADR  000052          LR.TYP  000002          MPCSR = 177746          OP$DUP= 000002          PIRQ  = 177772
LO.CLS  000002          LR.UCB  000016          MS.ALL= 000437 G        OP$EST= 000100          PMODE = 030000
LO.CON  000006          LR.UIC  000020          MS.CIR= 001000 G        OP$EVE= 010000          PREFIX  001622R
LO.COP  000010          LR.UNT  000005          MS.CSR= 000001 G        OP$FIL= 000200          PREF2   001631R
LO.COS  000020          LS.ALI= 000016          MS.EXT= 000200 G        OP$GRP= 000100          PR0   = 000000
LO.CSR  000010          LS.CEX= 000001          MS.INC= 000100 G        OP$HOS= 000001          PR1   = 000040
LO.CTL  000032          LS.CIR= 000033          MS.LLC= 000040 G        OP$HTM= 000002          PR4   = 000200
LO.DES  000002          LS.CXO= 010000          MS.MTP= 000010 G        OP$INA= 004000          PR5   = 000240
LO.DTE  000047          LS.CX5= 000400          MS.MUX= 000420 G        OP$INC= 001000          PR6   = 000300
LO.EVT  000004          LS.ECH= 001000          MS.PRI= 000002 G        OP$KDS= 002000          PR7   = 000340
LO.GRO  000024          LS.FDX= 004000          MS.SEC= 000400 G        OP$KDT= 000400          PS    = 177776
LO.HOS  000040          LS.HDX= 010000          MS.VCT= 000004 G        OP$KEV= 001000          PSTART  001002RG
LO.HTM  000032          LS.HLP= 000012          M$$CRB= 000124          OP$KGR= 001000          PUMR    001010RG
LO.INA  000062          LS.LIN= 000003          M$$CRX= 000000          OP$KNT= 010000          P$$P45= 000000
LO.INC  000060          LS.LMC= 000007          M$$FCS= 000000          OP$KSK= 002000          P$$WRD= 000000
LO.LDT  000046          LS.LOG= 000020          M$$MGE= 000000          OP$LCT= 000200          QF$ACT= 000001
LO.LID  000030          LS.MOD= 000036          M$$NET= 000000          OP$LNE= 000040          QF$ALL= 000002
LO.LIN  000004          LS.NOD= 000014          M$$OVR= 000000          OP$LOC= 000001          QF$KNO= 000004
LO.LTM  000034          LS.NTI= 000013          NAMES   001426R         OP$LST= 000002          QF$LOO= 000010
LO.MAC  000022          LS.OBJ= 000022          NONMAP  001000RG        OP$LTM= 000001          Q$$OPT= 000010
LO.MDE  000024          LS.OPT= 000400          NTPNAM  000734RG        OP$LVL= 000010          Q.IOAE= 000012 G
LO.NAM  000044          LS.PRO= 000002          NTPPAR  000740RG        OP$MAC= 000010          Q.IOEF= 000006
LO.NET= 000002          LS.TOP= 002000          NTPSAV  000756RG        OP$MCO= 000002          Q.IOFN= 000002 G
LO.NNM  000022          LS.TST= 000010          NTPSIZ  000750RG        OP$MDE= 000400          Q.IOLU= 000004
LO.OFL  000012          LS.UNF= 020000          N$$ACC= 000001          OP$MLI= 000004          Q.IOPL= 000014 G
LO.ONM  000004          LS.XIT= 000011          N$$BUF= 000001          OP$MON= 000100          Q.IOPR= 000007
LO.OUT  000056          LX.ALI= 000017          N$$LDV= 000001          OP$MST= 000020          Q.IOSB= 000010 G
LO.OWN  000026          LX.CEX= 000004          N$$MCP= 000001          OP$NAM= 000004          RA      000730RG
LO.PAR  000010          LX.CIR= 000034          N$$MLL= 000001          OP$NET= 004000          RB      000732RG
LO.PRI  000012          LX.LIN= 000006          N$$MOV= 000010          OP$NLI= 000002          RQB     003430RG
LO.RET  000064          LX.LOG= 000021          N$$NCT= 000001          OP$NOD= 000020          RQBST = 003430RG
LO.RPA  000030          LX.NOD= 000015          N$$PEM= 000001          OP$OUT= 002000          R$$DER= 000000
LO.SCR  000016          LX.OBJ= 000023          N.ANM1= 000002          OP$OWN= 000020          R$$K11= 000001
LO.SEG  000054          LX.PRO= 000005          N.ANM2= 000024          OP$PAR= 000010          R$$SND= 000000
LO.SKA  000014          L$$ASG= 000000          N.DID = 000024          OP$PRI= 000010          R$$11M= 000000
LO.SKN  000014          L$$DRV= 000000          N.DVNM= 000032          OP$RET= 010000          SBFLG   000772RG
LO.SNM  000035          L$$P11= 000001          N.FID = 000000          OP$RPA= 000010          SBNAM   000766RG
LO.TPA  000020          L$$11R= 000000          N.FNAM= 000006          OP$SCO= 000001          SBPAR   000762RG
LO.TRB  000034          L.COST  000015 G        N.FTYP= 000014          OP$SEG= 000400          SBPCB   000746RG
LO.TRI  000030          L.CTL   000012 G        N.FVER= 000016          OP$SER= 000400          SBPOOL  000760RG
LO.UNT  000033          L.CVA   177776 G        N.NEXT= 000022          OP$SIN= 000010          SF.ACT= 000200 G
```

```
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 11
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

$LGPDV     004122 RG        #17-848
$LGSTT     004130 RG        #17-851
$LGUIC     004176 RG        #17-863
$LLCTA     003710 RG        #17-763
$LOWSZ     000472 RG        #6-124        19-1091
$LSTAD     000724 RG        #7-170
$LSTRM     000722 RG        #7-169
$LTMFC     004010 RG        #17-802
$MAC       001616 RG        #13-398
$MAXOV     004120 RG         6-126       #17-847
$MISS      002014 RG        #15-460
$MXVCT     003604 RG        #16-707
$NAME      002344 RG        #15-516
$NBIAS     004072 RG        #17-829
$NMCLH     004212 RG        #17-868
$NMLST     004110 RG        #17-841
$NTLHB     004266 RG        #17-968
$NTLPT     004066 RG        #17-827
$NTUIC     004264 RG        #17-966
$OBJHD     004112 RG        #17-842
$OPTON     002004 RG        #15-456
$PANIC  =  ******    GX      14-446       22-1275
$PARHD     003700 RG        #16-753
$PAVL      004106 RG        #17-839
$PDVA      002342 RG        #15-515
$PDVNM     003712 RG        #17-764
$PDVTA     003704 RG        #17-761
$PFLAG     003416 RG        #15-622
$PRAVL     004260 RG        #17-941
$PRV.N     001266 RG        #11-344
$PSHPT     000174 RG        #6-102
$PSIPT     004032 RG        #17-813       25-1350
$PUMR      004100 RG        #17-834
$PUTAD     004700 RG        #20-1157
$PUTRC     004444 RG        #19-1058
$PUTRD     004444 RG        #19-1079
$PWRF1     004102 RG        #17-837
$QBIAS     004074 RG        #17-830
$QSTRT     004076 RG        #17-831
$QUAL      002010 RG        #15-458
$RADDR     000456 RG        #6-118        18-1021      19-1084      19-1095      21-1222      23-1285
$RBLCK     000460 RG        #6-119        21-1213      23-1285     *24-1313     *24-1322     *24-1323
$RDBAF     004060 RG        #17-824
$RDBCT     004052 RG        #17-821
$RDBLH     004054 RG        #17-822
$RDBNM     003722 RG        #17-768
$RDBSZ     003724 RG        #17-769
$RDBTH     003742 RG        #17-776
$RDOCT     004064 RG        #17-826
$RDOSL     004062 RG        #17-825
$REP.C     001222 RG        #10-309        11-342
$REP.N     001272 RG         11-343      #11-346
```

```
    196                                                   .SBTTL  ERROR MESSAGES
    197                                           ;
    198                                           ; ERROR MESSAGE BUFFER
    199                                           ;
    200 000370      015                           ERRMSG: .BYTE   15
    201 000371      126      116      120                 .ASCII  "VNP -- "
    202 000400                                    ERRBUF: .BLKB   70.
    203
    204                                                   .EVEN
    205
    206                                           ;
    207                                           ; ERROR MESSAGES
    208                                           ;
    209 000506                                            NTLER$  ,99,7,$RERR,REP7,,<Illegal function>
    210 000540                                            NTLER$  ,98,7,$RERR,REP7,,<Communications executive not loaded>
    211 000614                                            NTLER$  ,97,7,$RERR,REP7,,<Process not in system>
    212 000652                                            NTLER$  ,96,7,$RERR,REP7,,<Garbage in process descriptor vector>
    213 000730                                            NTLER$  ,95,7,$RERR,REP7,,<Line not in system>
    214 000764                                            NTLER$  ,94,7,$RERR,REP7,,<Illegal CSR address>
    215 001020                                            NTLER$  ,93,7,$RERR,REP7,,<Illegal interrupt priority>
    216 001064                                            NTLER$  ,92,7,$RERR,REP7,,<Illegal interrupt vector>
    217 001126                                            NTLER$  ,91,7,$RERR,REP7,,<No data block address>
    218 001164                                            NTLER$  ,90,7,$RERR,REP7,,<Illegal unit number>
    219 001220                                            NTLER$  ,88,7,$RERR,REP7,,<Garbage in DLC process descriptor vector>
    220 001302                                            NTLER$  ,87,7,$RERR,REP7,,<Units not in ascending order>
    221 001350                                            NTLER$  ,86,7,$RERR,REP7,,<Tributary not in system>
    222 001410                                            NTLER$  ,Z1,7,$RERR,REP7,,<Device offline>
    223 001440                                            NTLER$  ,Z3,7A,$RERR,REP7A,,<still active>
    224 001466                                            NTLER$  ,Z4,7,$RERR,DEV7,,<Device **: still mounted>
    225 001530                                            NTLER$  ,Z5,7B,$RERR,REP7B,,<loaded>
    226 001550                                            NTLER$  ,Z6,7,$RERR,REP7,,<Line in wrong state>
    227 001604                                            NTLER$  ,Z7,7,$RERR,REP7,,<Device **: still active>
    228 001644                                            NTLER$  ,Z8,7,$RERR,REP7,,<Node in wrong state>
    229 001700                                            NTLER$  ,ZB,7,$RERR,REP7,,<Line in wrong state>
    230 001734                                            NTLER$  ,ZP,7,$RERR,REP7,,<Vector not in system>
    231 001772                                            NTLER$  ,ZR,7,$RERR,REP7,,<Process may not be unloaded>
    232 002036                                            NTLER$  ,01,7,$RERR,REP7,,<XPT process not found>
    233 002074                                            NTLER$  ,02,7,$RERR,REP7,,<XPT process not loaded>
    234 002134                                            NTLER$  ,03,7,$RERR,REP7,,<Reverse mapping table inconsistent>
    235 002210                                            NTLER$  ,04,7,$RERR,REP7,,<Line not assigned to XPT>            ;*;
    236 002252                                            NTLER$  ,05,7,$RERR,REP7,,<Line parameter not applicable>       ;*;
    237 002320                                            NTLER$  ,06,7,$RERR,REP7,,<PSI not generated>
    238
    239                                           ;
    240                                           ;ERROR MESSAGE FORMAT STRINGS
    241                                           ;
    242 002352      000                           FMT7:   .ASCIZ  ""
    243 002353      103      150      141         FMT7A:  .ASCIZ  "Channel * "
    244 002366      114      151      156         FMT7B:  .ASCIZ  "Line * "
    245                                                   .EVEN
    246
    247 000000                                            .PSECT
```

```
001214  001415                         BEQ     121$            ; IF EQ, NO
001216  026702  000000G                CMP     $DDMPV,R2       ; COMBINATION DDM/DLC?
001222  001001                         BNE     30$             ; IF NE, NO
001224  005002                         CLR     R2              ; INDICATE NO DLC PDV
001226  010267  000000G       30$:     MOV     R2,$DLCPV       ; SAVE DLC PDV ADDRESS
001232                                 RETURN

                              ;
                              ; ERROR CONDITIONS
                              ;
001234                        101$:    EMSG$R  95              ; LINE NOT IN SYSTEM
001242                        111$:    EMSG$R  90              ; ILLEGAL UNIT NUMBER
001250                        121$:    EMSG$R  88              ; GARBAGE IN DLC PDV
001256                        131$:    EMSG$R  86              ; TRIBUTARY NOT IN SYSTEM
```

```
                                 .SBTTL  $FPDV - FIND PDV
                         ;+
                         ; $FPDV - TRY TO FIND PDV
                         ;
                         ; INPUTS:
                         ;       R0=RAD50 PDV NAME
                         ;
                         ; OUTPUTS:
                         ;       C-BIT=SUCCESS/FAILURE
                         ;       R0=PDV ADDRESS
                         ;       R1,R2=DESTROYED
                         ;-
002776                   $FPDV:
002776  010046                   MOV     R0,-(SP)        ; SAVE NAME TO BE MATCHED
003000  016701  000000G          MOV     $PDVTA,R1       ; GET PDV TABLE ADDRESS
003004  016702  000000G          MOV     $PDVNM,R2       ; AND NUMBER OF PDV'S
003010  001407                   BEQ     30$             ; IF ZERO, NO PDV'S!
003012  012100           10$:    MOV     (R1)+,R0        ; GET PDV ADDRESS
003014  001403                   BEQ     20$             ; IF ZERO, SKIP THIS ONE
003016  026016  000000G  15$:    CMP     Z.NAM(R0),(SP)  ; DOES THE NAME MATCH?
003022  001403                   BEQ     40$             ; IF EQ, YES (C-BIT CLEAR)
003024  005302           20$:    DEC     R2
003026  001371                   BNE     10$             ; LOOP $PDVNM TIMES
003030  000261           30$:    SEC                     ; INDICATE FAILURE
003032  005226           40$:    INC     (SP)+           ; PURGE STACK
003034  006016                   ROR     (SP)            ; SAVE C-BIT
003036  005741                   TST     -(R1)           ; CALCULATE PDV INDEX
003040  166701  000000G          SUB     $PDVTA,R1       ;    ...
003044  010167  000246'          MOV     R1,PDVX         ; SAVE PDV INDEX VALUE
003050  006316                   ASL     (SP)            ; RESTORE C-BIT
003052                           RETURN
```

```
                                                              J  1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 6
    BOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

HRDADD     014550 RG        #93-3988
HSAD       014404 RG        #91-3903
HSAR       014364 RG        #91-3897
IBUF       000044 RG        #19-1139     30-1798
IBUF0      000000 RG        #18-1108     30-1783
IBUF2      000124 RG        #20-1178     30-1811
IBUF3      000254 RG        #21-1220     30-1768
IBUF4      000306 RG        #22-1250     30-1771
IBUF5      000324 RG        #23-1265     30-1774
IE.EOF   = ******  GX        28-1710
INAC       010624 R         #67-3054
INCT       010500 R         #67-3035
INIBUF     016002 R         #99-4340     101-4390    103-4545    103-4549    103-4553
IO.WVB   = ******  GX        10-624
IPL        011022 R         #67-3081
ISTATE     000466 RG         31-1873    #32-1884
I$$AS    = ******            5-72        5-75
KBUF       000002 RG        #19-1139     30-1797
KBUF0      000000 RG        #18-1108     30-1782
KBUF2      000004 RG        #20-1178     30-1810
KBUF3      000014 RG        #21-1220     30-1767
KBUF4      000020 RG        #22-1250     30-1770
KBUF5      000022 RG        #23-1265     30-1773
KISAR0   = 172340           #5-75
KISAR6   = 172354           #5-75
KWORD      000032 RG         31-1872    #32-1884
LBRA     = 000074           #27-1666
LDDLM      017636 R          106-4753   #108-4808
LDLINE     017516 R          106-4756   #107-4777
LDPRO      020026 R          106-4738   #109-4849
LD.ALI   = 000030           #5-78
LD.CEX   = 000024           #5-78
LD.CIR   = 000035           #5-78
LD.LIN   = 000025           #5-78
LD.LOG   = 000031           #5-78
LD.MOD   = 000037           #5-78
LD.NOD   = 000027           #5-78
LD.OBJ   = 000032           #5-78
LD.PRO   = 000026           #5-78
LET      = 000100           #7-202       96-4140
LF.ACT   = 100000           #5-73
LF.BRO   = 000400           #5-73        59-2703
LF.BWT   = 000007           #5-73
LF.ENA   = 002000           #5-73        28-1695     59-2705     59-2714
LF.LPB   = 001000           #5-73
LF.MDC   = 000100           #5-73
LF.MFL   = 004000           #5-73        28-1695     59-2712     66-3002     106-4746
LF.MTP   = 000020           #5-73
LF.PAC   = 000200           #5-73
LF.RDY   = 040000           #5-73
LF.REA   = 010000           #5-73
LF.SER   = 000040           #5-73        28-1695
```

```
                                                              K  1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 7
```

```
                                                                J  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 19
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL    VALUE              REFERENCES

$RADDR    = ******  GX       *8-268       8-280      *8-287      *8-295       8-305      *8-329      *8-402      *8-418      *8-448
                             *8-457      *11-742     *12-797     *13-834     *14-879     *16-948     *17-1054    *26-1569    *26-1573
                             *26-1577    *26-1583    *26-1594    *26-1597    *26-1607    *26-1614    *61-2753    *61-2778    *62-2792
                             *62-2808    *63-2841    *63-2863    *64-2889    *65-2920    *65-2926    *65-2956    *65-2960    *67-3017
                             *67-3027    *67-3037    *67-3046    *67-3056    *67-3065    *67-3074    *67-3083    *67-3093    *67-3103
                             *67-3110    *69-3175    *69-3184    *69-3191    *70-3239    *71-3257    *72-3280    *73-3303    *74-3318
                             *75-3344    *75-3372    *76-3392    *76-3408    *79-3506    *80-3542    *86-3686    *87-3726    *88-3747
                             *88-3762    *94-4060    *94-4063    *94-4076    *98-4258    *98-4272    *102-4451   *102-4454   *102-4467
                             *103-4524   *103-4526   *103-4530   *103-4536   *105-4707
$RAD50    = 000016           #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$RBLCK    = ******  G        *8-390      *8-415      *8-445      *13-829     *14-874
$RDBCT    = ******  GX       *54-2608
$RDBLH    = ******  GX        13-830      13-831     *13-836     *13-837     *16-955     *16-956
$RDBNM    = ******  GX        13-822      53-2587     54-2606     54-2608
$RDBS       002644 RG         8-503      #13-819
$RDBSZ    = ******  GX        13-823      53-2588
$RDBTH    = ******  GX        53-2591
$RET1       001356 RG        #8-521
$RET2       001400 RG         8-522      #8-527
$RONLY    = ******            18-1108     18-1108     18-1108     19-1139     19-1139     19-1139     20-1178     20-1178     20-1178
                              21-1220     21-1220     21-1220     22-1250     22-1250     22-1250     23-1265     23-1265     23-1265
                              32-1884     32-1884     32-1884     46-2329     46-2329     46-2329

$RSIZE    = ******  GX       *8-267      *17-1043
$RSNTI      020156 RG        #110-4880
$SAVAL    = ******  GX        10-583      28-1682     106-4726    110-4877
$SAVRG    = ******  GX        108-4808
$SAVVR    = ******  GX        12-775
$SDBCT    = ******  GX       *54-2609
$SDBLH    = ******  GX        14-875      14-876     *14-881     *14-882     *16-958     *16-959
$SDBNM    = ******  GX        14-866      53-2589     54-2609
$SDBS       002776 RG         8-499      #14-863
$SDBSZ    = ******  GX        14-868      53-2590
$SLTA     = ******  GX       *59-2700     59-2709
$SLTMA    = ******  GX        28-1691     59-2683     66-2991     97-4215     106-4742
$SLTNM    = ******  GX        28-1692     59-2684     97-4220     106-4743
$STRNG    = 000004           #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$SUBXP    = 000010           #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$SWOVL    = ******  GX        8-519       8-525
$UMRAL      002262 R          8-362      #11-684
$VFLAG    = ******  GX       *8-523      *8-527       17-1083    *110-4878   *110-4881
$VNTI       000000 RG        #8-218
$VRQB     = ******  GX        110-4879
$XAVL     = ******  GX       *8-382      *8-389
$XBIAS    = ******  GX       *8-381      *8-388
$XLINK    = ******  GX        105-4713
$XPAR     = ******  GX        16-972      16-976
$$LINX    = ******  GX       *67-3019
$$NODC    = ******  GX       *67-3029
$$$ARG    = 000003           #10-624      10-624     #10-624      10-624     #10-624      10-624     #10-624      10-624      10-624
$$$FLG    = 177777           #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
$$$GLB    = ******            10-624
$$$KEY    = 177777           #18-1108    #19-1139    #20-1178    #21-1220    #22-1250    #23-1265    #32-1884    #46-2329
```

```
                                                                K  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 20
```

```
                        ;
                        ; PARTITION CONTROL BLOCK ADDRESS
                        ;
000252                  PCBA:   .BLKW   1

                        ;
                        ; PARTITION NAME
                        ;
000254                  $PNAME::.BLKW   2

                        ;
                        ; LOAD SIZE OF PROCESS
                        ;
000260                  $LDSIZ::.BLKW   1

                        ;
                        ; UNA Micro code error msg buffer
                        ;
000262                  MBUF:   .BLKB   80.


                        ;
                        ; BLOCK I/O BUFFER
                        ;
000000                          .PSECT  ..BUF,D
000000                  DSKBUF: .BLKW   1

000000                          .PSECT
```

```
    730                                                 .ENDC                                                   ;[MP01]
    731
    732 002236  010567  000252'                 80$:    MOV     R5,PCBA                 ; SET PCB ADDRESS
    733 002242  005767  000250'                 90$:    TST     ERROR                   ; DID ERROR OCCUR IN KERNEL MODE? (CLEARS C-BIT)
    734 002246  001401                                  BEQ     110$                    ; IF ZERO, NO
    735 002250  000261                          100$:   SEC                             ; INDICATE FAILURE
    736 002252                                  110$:   RETURN
    737
    738                                         ;
    739                                         ; ERROR CONDITIONS
    740                                         ;
    741 002254  012767  000022  000250' 120$:   MOV     #ERR18,ERROR            ; PARTITION NOT IN SYSTEM
    742 002262  000767                                  BR      90$                     ;
    743 002264  012767  000024  000250' 130$:   MOV     #ERR20,ERROR            ; PROCESS/PARTITION BASE ADDRESS MISMATCH
    744 002272  000763                                  BR      90$                     ;
    745 002274  012767  000026  000250' 140$:   MOV     #ERR22,ERROR            ; PARTITION TOO SMALL
    746 002302  000757                                  BR      90$                     ;
    747 002304  012767  000030  000250' 150$:   MOV     #ERR24,ERROR            ; NOT COMMON PARTITION
    748 002312  000753                                  BR      90$                     ;
    749 002314  012767  000032  000250' 160$:   MOV     #ERR26,ERROR            ; PARTITION BUSY
    750 002322  000747                                  BR      90$                     ;
    751 002324  012767  000034  000250' 170$:   MOV     #ERR28,ERROR            ; ILLEGAL COMM EXEC PARTITION
    752 002332  000743                                  BR      90$                     ;
```

```
**FILE**ID**VPRMPT









VV      VV   PPPPPPPP     RRRRRRRR     MM      MM   PPPPPPPP     TTTTTTTTTT
VV      VV   PPPPPPPP     RRRRRRRR     MM      MM   PPPPPPPP     TTTTTTTTTT
VV      VV   PP      PP   RR      RR   MMMM  MMMM   PP      PP       TT
VV      VV   PP      PP   RR      RR   MMMM  MMMM   PP      PP       TT
VV      VV   PP      PP   RR      RR   MM  MM  MM   PP      PP       TT
VV      VV   PP      PP   RR      RR   MM  MM  MM   PP      PP       TT
VV      VV   PPPPPPPP     RRRRRRRR     MM      MM   PPPPPPPP         TT
VV      VV   PPPPPPPP     RRRRRRRR     MM      MM   PPPPPPPP         TT
VV      VV   PP           RR  RR       MM      MM   PP               TT
VV      VV   PP           RR  RR       MM      MM   PP               TT
  VV  VV     PP           RR    RR     MM      MM   PP               TT           ....
  VV  VV     PP           RR    RR     MM      MM   PP               TT           ....
    VV       PP           RR      RR   MM      MM   PP               TT           ....
    VV       PP           RR      RR   MM      MM   PP               TT           ....






LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
      54                                        .SBTTL  MACRO DEFINITIONS
      55                                ;***
      56                                ; LIBRARY MACROS
      57                                ;***
      58                                        .MCALL  ASL$,EMSG$R,GETRV,PUTRC,GETAD,PUTAD,CTRDF$,XPDDB$,SLTDF$
      59                                        .MCALL  BIAS,ASR$,ECDDB$,DHBDF$,RESRG,SAVRG,CACDF$,ADJDF$
      60                                        .MCALL  SWTCHI,SWTCHO,VECDF$,NTLDF$,CEACC$
      61
      62                                        .IF DF  R$$MPL                                                  ;[MP01]
      63                                        .MCALL  CTBDF$                                                  ;[MP01]
      64                                        CTBDF$                                                          ;[MP01]
      65                                        .ENDC                                                           ;[MP01]
      66
      67
      68 000000                                 CTRDF$                  ; DEFINE ECL COUNTERS OFFSETS AND SYMBOLS
      69 000000                                 ADJDF$                  ; DEFINE ADJACENCY DATA BASE SYMBOLS
      70 000000                                 ECDDB$                  ; DEFINE ECL DATA BASE OFFFSETS
      71 000000                                 XPDDB$                  ; DEFINE XPT'S DDB OFFSETS AND SYMBOLS
      72 000000                                 SLTDF$                  ; DEFINE SLT OFFSETS AND SYMBOLS
      73 000000                                 DHBDF$                  ; DEFINE DECNET HOME BLOCK OFFSETS
      74 000000                                 CACDF$                  ; DEFINE CACHE OFFSETS
      75 000000                                 VECDF$                  ; DEFINE VECTOR BLOCK OFFSETS
      76 000000                                 NTLDF$  ,,H             ; DEFINE NTL HOME BLOCK OFFSETS
      77
      78                                        .ENABL  LC
```

```
    657 001660  066703  000026G                 ADD     N$ROA1+4+$BFR,R3          ; GET START OF NODE ROL TABLE
    658 001664  010367  000034G                 MOV     R3,N$ROA2+4+$BFR          ; SET VIRTUAL ADDRESS OF AREA ROL
    659
    660 001670                          20$:    PUTRC                             ; WRITE XPT DDB TO IMAGE
    661
    662 001674  060504                          ADD     R5,R4                     ; DETERMINE TOTAL NUMBER OF ENTRIES
    663 001676  005204                          INC     R4                        ; ALLOW ONE MORE ENTRY FOR NODE ROL
    664 001700  005705                          TST     R5                        ; MULTI AREA ?
    665 001702  001401                          BEQ     30$                       ; IF NO - BRANCH
    666
    667 001704  005204                          INC     R4                        ; ALLOW ONE MORE ENTRY FOR AREA ROL
    668 001706  006304                  30$:    ASL     R4                        ; CONVERT TO BYTES
    669 001710  010401                          MOV     R4,R1                     ; COPY ROL SIZE
    670
    671 001712  022704  001000          40$:    CMP     #512.,R4                  ; MORE THAN ONE BLOCK ?
    672 001716  002002                          BGE     50$                       ; IF NO - BRANCH
    673 001720  012704  001000                  MOV     #512.,R4                  ; SET SIZE TO ONE BLOCK
    674
    675 001724  160401                  50$:    SUB     R4,R1                     ; CLACULATE REMAINING SIZE
    676 001726  010405                          MOV     R4,R5                     ; SAVE TRANSFER SIZE
    677 001730  012703  000000G                 MOV     #$BFR,R3                  ; STORE BUFFER ADDRESS
    678 001734  105023                  60$:    CLRB    (R3)+                     ; INIT BUFFER
    679 001736                                  SOB     R5,60$                    ; ...
    680
    681 001742                                  PUTAD   R2,R4,R0                  ; WRITE ROL TO IMAGE
    682 001762  060402                          ADD     R4,R2                     ; UPDATE VIRTUAL ADDRESS
    683 001764  010104                          MOV     R1,R4                     ; COPY REMAINING SIZE
    684 001766  001351                          BNE     40$                       ; IF YES - BRANCH
    685
    686 001770                          70$:    RETURN
    687
    688
    689 001772                          80$:    EMSG$R  99                        ; ROL ALLOCATION FAILURE
    690
```

```
                                SWTCHO  #$CTLST         ; SET BUFFER
                                PUTRC   $CTBIM,#2       ; UPDATE CTB POINTER IN IMAGE
                                SWTCHO                  ; REST BUFFER
                                MOV     R1,$CTLST       ;       ...
                        ;
                        ; SEARCH DEVICE LIST FOR RE-CONFIGURATION DEVICE
                        ;
                                MOV     $DEVHD,$BFR     ; GET START OF DEVICE LIST
                        10$:    MOV     $BFR,R0         ; GET NEXT DCB ADDRESS
                                BEQ     101$            ; IF EQ, END OF LIST
                                GETRV   R0,#D.PCB+2     ; READ NEXT DCB
                                CMP     $RCDEV,$BFR+D.NAM ; IS THIS THE RE-CONFIGURATION DEVICE ?
                                BNE     10$             ; IF NE, NO .. KEEP LOOKING

                                MOV     CTBSAV,R1       ; GET CTB ADDRESS
                                MOV     R0,L.DCB(R1)    ; STORE DCB ADDRESS IN CTB
                        20$:    CLC                     ; SUCCESS
                                RETURN                  ; RETURN TO USER MODE

                        ;
                        ; ERROR CONDITIONS
                        ;
                        101$:   EMSG$R  9A              ; ERROR - COULD NOT FIND RE-CONFIGURATION DEV.

                                .ENDC


        000001                  .END
```

Table of contents

```
     5-   63    MACRO DEFINITIONS
     6-   84    DATA
     8-  218    $AZFS - SET UP ASCIZ FILE SPEC
     9-  262    SETUP DEFAULT FILE NAME
    10-  289    $REP.C,$REP.P - REPLACE FILE NAME
    11-  326    $REP.N - REPLACE ERROR CODE
    11-  327    $CUR.N,$PRV.N - REPLACE FILE NAME AND ERROR CODE
    12-  359    SLFNP - SETUP LOCAL FILE NAME POINTER
    13-  380    $ASFS DATA
    14-  402    ERROR HANDLING
    15-  451    DATA FOR INPUT PARAMETERS
    16-  644    RSX DATA FROM RSX11M.STB
    17-  756    CEX DATA
    18-  971    $GETRC,$GETRV - GET A RECORD FORM SYSTEM IMAGE FILE
    19- 1038    $PUTRC - PUT RECORD TO SYSTEM IMAGE FILE
    20- 1140    $PUTAD - PUT RECORD TO SYSTEM IMAGE FILE
    21- 1167    $GETAD - GET RECORD FROM SYSTEM IMAGE FILE
    22- 1239    $VGCL - GET COMMAND LINE
    23- 1277    ERROR MESSAGES
    24- 1294    $CEACC - ACCESS BLOCK IN EXTENDED POOL
    25- 1327    $SWOVL - SWITCH OVERLAYS
```

```
                                                .SBTTL  $ASFS DATA
                                        ;
                                        ; $AZFS DATA AREA
                                        ;

                                                .NLIST BEX
     001362  000000                     FNPTR:  .WORD   0               ; POINTER TO FILENAME
     001364     015                     FNERR:  .BYTE   15
     001365     126     116     120             .ASCII  "VNP -- "
     001374     123     131     072     FILE::  .ASCIZ  "SY:[XXX,XXX]XXXXXXXXX.XXX"
     001426                             NAMES:  .BLKB   7*10.
     001534     123     131     060     $DFUIC::.ASCIZ  "SY0:[XXX,XXX]"
     001552     130     130     130     $DFNAM::.ASCIZ  "XXXXXXXXX"
     001564     103     105     124     $CET::  .ASCIZ  "CETAB "
     001573     122     123     130     $RSX::  .ASCIZ  "RSX11M"
     001602     103     105     130     $CEX::  .ASCIZ  "CEX"
     001606     124     123     113     DFEXT:  .ASCIZ  "TSK"
     001612     123     124     102     $STB::  .ASCIZ  "STB"
     001616     115     101     103     $MAC::  .ASCIZ  "MAC"
                                                .EVEN
                                                .NLIST BEX
```

```
 004420 001002                          BNE     10$             ; IF NE YES
 004422 012716  100000                  MOV     #100000,(SP)    ; SET LIMIT FOR MAPPED SYSTEM
 004426 022626                  10$:    CMP     (SP)+,(SP)+     ; VALID ADDRESS?
 004430 103505                          BLO     30$             ; IF LO NO

 004432                         $GETRC::
 004432 004267  000000G         $GETRD::JSR     R2,$SAVVR       ; SAVE VOLATILE REGISTERS
 004436 012746  000000G                 MOV     #.GET,-(SP)     ; SET ADDRESS OF RECORD ACCESS ROUTINE
 004442 000404                          BR      20$
```

```
                                                                J 12
VRES - VNP ROOT MACRO V05.03b 1uesday 17-Sep-85 16:15  Page 26-3
Symbol table

SF.ENA= 000100 G    UISDR0= 177600      Z.NAM   000004 G            $DDFNC  004036RG    $PANIC= ****** GX
SF.LPB= 000004 G    UISDR4= 177610      Z.PCB   000012 G            $DDMPV  002420RG    $PARHD  003700RG
SF.MFL= 000040 G    UISDR5= 177612      Z.SCH   000007 G            $DECPT  004030RG    $PAVL   004106RG
SF.PAC= 000020 G    UISDR6= 177614      $ADRX   000504RG            $DEVHD  003672RG    $PDVA   002342RG
SF.REA= 000010 G    UISDR7= 177616      $ADRZ   000502RG            $DFN    001160RG    $PDVNM  003712RG
SF.SER= 000001 G    UMRFLG  000776RG    $ASTLV  000210RG        003 $DFNAM  001552RG    $PDVTA  003704RG
SF.SVC= 000002 G    UMRPTR  000774RG    $AZFS   001042RG            $DFN2   001162RG    $PFLAG  003416RG
SF.UNL= 000040 G    VF.ACP= 001000 G    $BFR    000000RG        004 $DFUIC  001534RG    $PRAVL  004260RG
SISDR0= 172200      VF.ADD= 000400 G    $BIAS   002346RG            $DLCPV  002416RG    $PRV.N  001266RG
SLFNP   001332R     VF.DLX= 004000 G    $BIGCT  000726RG            $ERMSG= ****** GX   $PSHPT  000174RG        003
SLTNUM  001040RG    VF.EAD= 020000 G    $CBDSG= ****** GX           $EROUT  001674RG    $PSIPT  004032RG
SPARE1  004220RG    VF.ECH= 000020 G    $CCBAF  003734RG            $EVDSC  004132RG    $PUMR   004100RG
SPARE2  004224RG    VF.FIX= 000100 G    $CCBAL  003740RG            $EXSIZ  003674RG    $PUTAD  004700RG
SPSAV   001006RG    VF.INI= 000040 G    $CCBCT  003732RG            $FEATR  002020RG    $PUTRC  004444RG
SR0    = 177572     VF.NTI= 000001 G    $CCBLH  004040RG            $FILHD  004126RG    $PUTRD  004444RG
SR3    = 172516     VF.ON  = 000002 G   $CCBNM  003716RG            $FLAGS  003406RG    $PWRF1  004102RG
SUBPAR  000744RG    VF.PEM= 002000 G    $CCBSZ  003720RG            $FMASK  003676RG    $QBIAS  004074RG
SWR    = 177570     VF.PSI= 040000 G    $CEACC  005622RG            $FNOUT  001642RG    $QSTRT  004076RG
SYSDNB  000416R     VF.SED= 100000 G    $CEAVL  004236RG            $FPDV = ****** GX   $QUAL   002010RG
SYSFDB  000256RG    VF.SKA= 010000 G    $CELFN  004210R             $FUNC   002002RG    $RADDR  000456RG
S$$WRG= 000000      VF.SS  = 000010 G   $CET    001564RG            $GCLEP= ****** GX   $RBLCK  000460RG
S$$YSZ= 007600      VF.TRA= 000004 G    $CETAB  000000RG        006 $GENND  000720RG    $RDBAF  004060RG
S.ANM1= 000014      VF.VDT= 000200 G    $CETDF  004242RG            $GENST  000716RG    $RDBCT  004052RG
S.ANM2= 000005      V$$CTR= 001000      $CETLN  004244RG            $GETAD  004712RG    $RDBLH  004054RG
S.BFHD= 000020      V1.NSY= 000001 G    $CETND  000000RG        007 $GETRC  004432RG    $RDBNM  003722RG
S.COST  000001 G    WC.CNT= 000400      $CETST  004246RG            $GETRD  004432RG    $RDBSZ  003724RG
S.ETEF= 000002      WC.EVE= 004000      $CEX    001602RG            $GETRV  004366RG    $RDBTH  003742RG
S.FATT= 000016      WC.TRI= 002000      $CEXND  004234RG            $HMBUF  000612RG    $RDQCT  004064RG
S.FDB  = 000140     WC.UNT= 001000      $CLASN  000162RG        003 $ICBI = 000016 G    $RDQSL  004062RG
S.FLG   000000 G    W.TSEF= 000002      $CLAST  000172RG        003 $ICBLN  002016RG    $REP.C  001222RG
S.FNAM= 000006      X$$DBT= 000000      $CLBUF= 000310RG        003 $KRBLN= 000004 G    $REP.N  001272RG
S.FNB  = 000036     ZF.COU= 001000 G    $CLCSI  000212RG        003 $LDBAF  003736RG    $REP.P  001226RG
S.FNBW= 000017      ZF.DDM= 000001 G    $CLEVL  000206RG        003 $LGCON  004204RG    $RET1 = ****** GX
S.FNTY= 000004      ZF.DIA= 004000 G    $CLFDB  000266RG        003 $LGDDB  004124RG    $RET2 = ****** GX
S.FTYP= 000002      ZF.DLC= 000002 G    $CLFLG  000464RG        003 $LGFNB  004152RG    $RSIZ   000454RG
S.LEN   000004 G    ZF.DVP= 100000 G    $CLIOS= 000132RG        003 $LGMON  004200RG    $RSIZE= 000260RG
S.MPR   003550R     ZF.INI= 040000 G    $CLMCR  000000RG        003 $LGPDV  004122RG    $RSX    001573RG
S.NFEN= 000020      ZF.KMX= 000020 G    $CLPMT  000466RG        003 $LGSTT  004130RG    $SAVVR= ****** GX
S.NMST  000002 G    ZF.LLC= 000004 G    $CLPRM  000472R         003 $LGUIC  004176RG    $SDBAF  004050RG
S.OWNR  000003 G    ZF.LMC= 000100 G    $CLQIO  000122RG        003 $LLCTA  003710RG    $SDBCT  004042RG
TIOUT   000000R     ZF.MAN= 020000 G    $CLRTN= ****** GX           $LOWSZ  000472RG    $SDBLH  004044RG
TOPFLG  000754RG    ZF.MFL= 000010 G    $CLSBK  000176RG        003 $LSTAD  000724RG    $SDBNM  003726RG
TPS    = 177564     ZF.MTM= 000400 G    $CLSEF  000156RG        003 $LSTRM  000722RG    $SDBSZ  003730RG
TY$CHA= 000400      ZF.MUX= 000040 G    $CLSIZ= 000306RG        003 $LTMFC  004010RG    $SHLST  004216RG
TY$EVE= 001000      ZF.PSE= 002000 G    $CLSRV  000470RG        003 $MAC    001616RG    $SLN    002422RG
TY$STA= 002000      ZF.SLI= 010000 G    $CLTBF  000202RG        003 $MAXOV  004120RG    $SLTA   002414RG
TY$SUM= 000000      ZF.TIM= 000200 G    $CLTSZ  000204RG        003 $MISS   002014RG    $SLTMA  003706RG
T$$KMG= 000000      ZF.X3P= 000000 G    $CLWEF  000152RG        003 $MXVCT  003604RG    $SLTNM  003714RG
T$$MIN= 000000      ZS.ASN= 100000 G    $CMAND  002006RG            $NAME   002344RG    $SNAPT  004034RG
UBMPR  = 170200     ZS.BSY= 140000 G    $CMFRK  003752RG            $NBIAS  004072RG    $SQRCM  004116RG
UDSAR0= 177660      Z.AVL   000014 G    $CMPDV  003744RG            $NMCLH  004212RG    $STB    001612RG
UDSDR0= 177620      Z.DAT   000016 G    $CPBIA  000474RG            $NMLST  004110RG    $STMFC  003770RG
UISAR0= 177640      Z.DSP   000000 G    $CRAVL  004252RG            $NTLHB  004266RG    $SWOVL  005716RG
UISAR4= 177650      Z.FLG   000010 G    $CTIM   002364RG            $NTLPT  004066RG    $SYMND  003672RG
UISAR5= 177652      Z.LEN  = 000016 G   $CUR.N  001260RG            $NTUIC  004264RG    $SYMTB  003550RG
UISAR6= 177654      Z.LLN   000006 G    $CXOPT  004070RG            $OBJHD  004112RG    $SYSIZ  003702RG
UISAR7= 177656      Z.MAP   000020 G    $C5TA = ****** GX           $OPTON  002004RG    $TCK50  004114RG
```

K 12

```
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16     PAGE 12
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE             REFERENCES

$REP.P     001226 RG      #10-312     11-345
$RET1    = ******  GX      25-1354
$RET2    = ******  GX      25-1356
$RSIZ      000454 RG      #6-116
$RSIZE   = 000260 RG      #6-117      18-1024     23-1292
$RSX       001573 RG      #13-394
$SAVVR   = ******  GX      18-1034    19-1079     20-1163     21-1199     22-1253
$SDBAF     004050 RG      #17-820
$SDBCT     004042 RG      #17-817
$SDBLH     004044 RG      #17-818
$SDBNM     003726 RG      #17-770
$SDBSZ     003730 RG      #17-771
$SHLST     004216 RG      #17-915
$SLN       002422 RG      #15-541
$SLTA      002414 RG      #15-538
$SLTMA     003706 RG      #17-762
$SLTNM     003714 RG      #17-765
$SNAPT     004034 RG      #17-814
$SQRCM     004116 RG      #17-845
$STB       001612 RG      #13-397
$STMFC     003770 RG      #17-793
$SWOVL     005716 RG      #25-1346
$SYMND     003672 RG      #16-742
$SYMTB     003550 RG      #16-655
$SYSIZ     003702 RG      #16-754
$TCK50     004114 RG      #17-844
$TIOUT     001654 RG      #14-417     14-440
$TK100     004114 RG      #17-843
$TMPBF     000000 RG      #7-203
$TMPND     000000 RG      #7-208
$TRIB      002340 RG      #15-513
$TYPE      002012 RG      #15-459
$T1SCL     004012 RG      #17-803
$T100C     003772 RG      #17-794
$T100Q     003764 RG      #17-790
$T50CL     003772 RG      #17-795
$T50Q      003764 RG      #17-791
$UNBAD     000510 RG      #6-136
$UNBLK     000512 RG      #6-137
$UNBSZ     000506 RG      #6-135
$UNEND     000612 RG      #6-138
$VACP    = ******  GX      25-1355
$VCTOF     000464 RG      #6-121      19-1094
$VCTVB     000462 RG      #6-120      19-1093
$VFLAG     003412 RG      #15-589
$VFLG1     003414 RG      #15-620
$VGCL      005022 RG      #22-1253    22-1273
$XAVL      004106 RG      #17-840
$XBIAS     004104 RG      #17-838     24-1322
$XBUF2     003420 RG      #15-629
$XFRAD     000466 RG      #6-122      19-1089
$XFRSV     000470 RG      #6-123
```

```
                                                .SBTTL  $NLRQB - PROCESS REQUEST BLOCK
                                        ;+
                                        ; $NLRQB - PROCESS RECEIVE DATA BLOCK
                                        ;
                                        ; THIS ROUTINE IS CALLED TO PROCESS THE NETLDR REQUEST BLOCK AND TO
                                        ; INITIALIZE THE LOCATIONS "$FUNC" AND "$MISS".
                                        ;
                                        ; INPUTS:
                                        ;       $$LRB=LOAD REQUEST BLOCK
                                        ;
                                        ; OUTPUTS:
                                        ;       C-BIT=SUCCESS/FAILURE
                                        ;       ALL REGISTERS=DESTROYED
                                        ;       $FUNC=FUNCTION CODE (0=NOP)
                                        ;       $MISS=MISSING PARAMETERS BIT MASK
                                        ;       $PDVA=PDV ADDRESS (SET/CLEAR PROCESS ONLY)
                                        ;       $SLTA=SLT ADDRESS (SET/CLEAR LINE ONLY)
                                        ;-
000000                                  $NLRQB::
000000  010667  000250'                         MOV     SP,SPSAV        ; SAVE STACK POINTER
000004  012703  000004G                         MOV     #$$LRB+4,R3     ; GET REQUEST BLOCK ADDRESS
000010  012304                                  MOV     (R3)+,R4        ; GET FUNCTION CODE
000012  042704  177400                          BIC     #^C<377>,R4     ; ...
000016  001435                                  BEQ     101$            ; IF ZERO, ILLEGAL FUNCTION
000020  022704  000037                          CMP     #FMAX,R4        ; IS IT A VALID FUNCTION CODE?
000024  103432                                  BLO     101$            ; IF LO, NO
000026  010405                                  MOV     R4,R5           ; COPY FUNCTION CODE
000030  006305                                  ASL     R5              ; MAKE A WORD INDEX
000032  016505  177776'                         MOV     FTAB-2(R5),R5   ; GET SUBROUTINE TABLE POINTER
000036  001425                                  BEQ     101$            ; IF ZERO, ILLEGAL FUNCTION
000040  016767  000002G 000000G                 MOV     $FLAGS+2,$FLAGS ; INIT FLAG WORD WITH DEFAULT VALUES
000046  042767  000000C 000000G                 BIC     #^C<FL.TOP!FL.ECH!FL.WRN>,$FLAGS ; BE SURE ...
000054  005067  000000G                         CLR     $MISS           ; NO PARAMETERS MISSING YET (CLEARS C-BIT)
000060                                  10$:    CALL    @(R5)+          ; KEEP CALLING SUBROUTINES
000062  103402                                  BCS     20$             ; UNTIL C-BIT IS SET
000064  005715                                  TST     (R5)            ; OR END OF TABLE (0) IS REACHED
000066  001374                                  BNE     10$             ; ...
000070  010467  000000G                 20$:    MOV     R4,$FUNC        ; SET FUNCTION CODE
000074                                          SWTCHI                  ; BE SURE WE ARE USING THE
000102                                          SWTCHO                  ;  DEFAULT BUFFERS
000110                                          RETURN

                                        ;
                                        ; ERROR CONDITION
                                        ;
000112                                  101$:   EMSG$R  99              ; ILLEGAL FUNCTION CODE
```

```
                                          .SBTTL  CHKLP - CHECK LINE PARAMETERS
                                  ;+
                                  ; CHKLP - CHECK LINE PARAMETERS
                                  ;
                                  ; INPUTS:
                                  ;       LS.OPT SET IF ANY OPTIONAL PARAMETERS
                                  ;       R3=REQUEST BLOCK POINTER
                                  ;               .BLKW      1      ; LO.CSR - CSR ADDRESS
                                  ;               .BLKW      1      ; LO.PRI - INTERRUPT PRIORITY
                                  ;               .BLKW      1      ; LO.VCT - INTERRUPT VECTOR
                                  ;               .BLKW      1      ; LO.SCR - SECONDARY CSR
                                  ;
                                  ; OUTPUTS:
                                  ;       C-BIT=SUCCESS/FAILURE
                                  ;       R3=DESTROYED
                                  ;       $$CSR=CSR ADDRESS
                                  ;       $$PRI=INTERRUPT PRIORITY
                                  ;       $$VECT=INTERRUPT VECTOR
                                  ;       ..SCSR=SECONDARY CSR
                                  ;       $MISS MAY BE SET NON-ZERO IF ANY FIELDS ARE MISSING
                                  ;-

001264                            CHKLP:
001264  012767  000000G 000000G   10$:    MOV     #MS.ALL,$MISS     ; ALL PARAMETERS MISSING
001272  032767  000400  000004G           BIT     #LS.OPT,$$LRB+4   ; ANY OPTIONAL PARAMETERS INCLUDED?
001300  001002                            BNE     20$               ; IF NE, YES
001302  012703  000234'                   MOV     #NULL,R3          ; IF NOT, USE NULL INPUT PARAMETERS
001306  012300                    20$:    MOV     (R3)+,R0          ; GET CSR VALUE
001310  001411                            BEQ     30$               ; IF ZERO, FIELD MISSING
001312  022700  160000                    CMP     #160000,R0        ; IS IT AN I/O PAGE ADDRESS?
001316  101071                            BHI     101$              ; IF HI, NO
001320  032700  000003                    BIT     #3,R0             ; IS IT A MULTIPLE OF 4?
001324  001066                            BNE     101$              ; IF NE, NO
001326  042767  000000G 000000G           BIC     #MS.CSR,$MISS     ; INDICATE CSR INCLUDED
001334  010067  000000G           30$:    MOV     R0,$$CSR          ; SET CSR ADDRESS
001340  012300                            MOV     (R3)+,R0          ; GET PRIORITY VALUE
001342  001406                            BEQ     40$               ; IF ZERO, FIELD MISSING
001344  022700  000007                    CMP     #7,R0             ; IS IT RANGE 1-7?
001350  103457                            BLO     111$              ; IF LO, NO
001352  042767  000000G 000000G           BIC     #MS.PRI,$MISS     ; INDICATE PRIORITY INCLUDED
001360                            40$:    ASL$    5,R0              ; CONVERT TO PSW PRIORITY VALUES
001372  010067  000000G                   MOV     R0,$$PRI          ; SET INTERRUPT PRIORITY
001376  012300                            MOV     (R3)+,R0          ; GET VECTOR VALUE
001400  001411                            BEQ     50$               ; IF ZERO, FIELD MISSING
001402  032700  000003                    BIT     #3,R0             ; IS IT A MULTIPLE OF 4?
001406  001043                            BNE     121$              ; IF NE, NO
001410  026700  000000G                   CMP     $MXVCT,R0         ; IS IT A LEGAL VECTOR ?
001414  101443                            BLOS    131$              ; .. NO
001416  042767  000000G 000000G           BIC     #MS.VCT,$MISS     ; INDICATE VECTOR INCLUDED
001424  010067  000000G           50$:    MOV     R0,$$VECT         ; SET INTERRUPT VECTOR
001430  012300                            MOV     (R3)+,R0          ; GET SECONDARY CSR VALUE
001432  001411                            BEQ     60$               ; IF ZERO, FIELD MISSING
001434  022700  160000                    CMP     #160000,R0        ; IS IT AN I/O PAGE ADDRESS?
001440  101020                            BHI     101$              ; IF HI, NO
001442  032700  000003                    BIT     #3,R0             ; IS IT A MULTIPLE OF 4?
001446  001015                            BNE     101$              ; IF NE, NO
001450  042767  000000G 000000G           BIC     #MS.SEC,$MISS     ; INDICATE SECONDARY CSR INCLUDED
```

```
                                                .SBTTL  $FSLT - FIND SLT
                                        ;+
                                        ; $FSLT - TRY TO FIND SLT
                                        ;
                                        ; INPUTS:
                                        ;       R1=CONTROLLER NUMBER (BYTE)
                                        ;       R2=UNIT NUMBER (BYTE)
                                        ;       PDVX=DDM PDV INDEX
                                        ;
                                        ; OUTPUTS:
                                        ;       C-BIT=SUCCESS/FAILURE
                                        ;       R0=SLT ADDRESS
                                        ;       R1=SYSTEM LINE NUMBER
                                        ;       R2=NUMBER OF UNITS ON THIS CONTROLLER
                                        ;-
003054                                  $FSLT::
003054  012746  177777                          MOV     #-1,-(SP)       ; LEAVE ROOM FOR SLT ADDRESS AND SLN
003060  011646                                  MOV     (SP),-(SP)      ;   ...
003062  110146                                  MOVB    R1,-(SP)        ; SAVE CONTROLLER NUMBER
003064  110266  000001                          MOVB    R2,1(SP)        ; AND UNIT NUMBER
003070  016700  000000G                         MOV     $SLTMA,R0       ; GET SLT MAPPING TABLE ADDRESS
003074  012701  177777                          MOV     #-1,R1          ; GET FIRST LINE NUMBER
003100  005067  000244'                         CLR     UNIT            ; NO UNITS ON THIS CONTROLLER YET
003104  012002                          10$:    MOV     (R0)+,R2        ; GET NEXT SLT ADDRESS
003106  005201                                  INC     R1              ; AND NEXT LINE NUMBER
003110  026701  000000G                 20$:    CMP     $SLTNM,R1       ; IS IT END OF SLT'S?
003114  001436                                  BEQ     30$             ; IF EQ, YES
003116  126267  000000G 000246'                 CMPB    L.DDM(R2),PDVX  ; DOES DDM PDV INDEX MATCH?
003124  001367                                  BNE     10$             ; IF NE, NO
003126  126216  000000G                         CMPB    L.CTL(R2),(SP)  ; DOES CONTROLLER NUMBER MATCH?
003132  001364                                  BNE     10$             ; IF NE, NO
003134  126267  000000G 000244'                 CMPB    L.UNT(R2),UNIT  ; ARE UNIT NUMBERS IN ASCENDING ORDER?
003142  001035                                  BNE     101$            ; IF NE, NO
003144  016746  000244'                         MOV     UNIT,-(SP)      ; SAVE UNIT NUMBER
003150  062716  000002G                         ADD     #$SLN+2,(SP)    ; POINT TO SYSTEM LINE NUMBER
003154  110136                                  MOVB    R1,@(SP)+       ; SAVE MUX LINE NUMBER
003156  005267  000244'                         INC     UNIT            ; TALLY UNITS ON THIS CONTROLLER
003162  126266  000000G 000001                  CMPB    L.UNT(R2),1(SP) ; DOES UNIT NUMBER MATCH?
003170  001345                                  BNE     10$             ; IF NE, NO
003172  010266  000002                          MOV     R2,2(SP)        ; SAVE SLT ADDRESS
003176  010166  000004                          MOV     R1,4(SP)        ; SAVE SYSTEM LINE NUMBER
003202  032767  000000G 000000G                 BIT     #FL.MUX,$FLAGS  ; MUX DEVICE?
003210  001335                                  BNE     10$             ; IF NE, YES - SEARCH ENTIRE SLT
003212  005726                          30$:    TST     (SP)+           ; PURGE STACK
003214  012600                                  MOV     (SP)+,R0        ; RETRIVE SLT ADDRESS
003216  012601                                  MOV     (SP)+,R1        ; SYSTEM LINE NUMBER
003220  100404                                  BMI     35$             ; IF MI, LINE NOT FOUND
003222  016702  000244'                         MOV     UNIT,R2         ; RETURN UNITS FOUND IN R2
003226  005702                                  TST     R2              ; ANY UNITS FOUND? (CLEARS C-BIT)
003230  001001                                  BNE     40$             ; IF NE, SUCCESS
003232  000261                          35$:    SEC                     ; INDICATE FAILURE
003234                                  40$:    RETURN

                                        ;
                                        ; ERROR CONDITION
                                        ;
003236  016706  000250'                 101$:   MOV     SPSAV,SP        ; RESTORE SP TO $NLRQB ENTRY VALUE
```

K 1

```
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 7
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE              REFERENCES

LF.TIM  = 000010           #5-73
LF.UNL  = 020000           #5-73
LF.X2P  = 000000           #5-73
LINSIZ  = ******  GX        9-558      *30-1756    30-1843    *31-1869
LN.CLO  = 000000           #5-73
LN.DUM  = 000005           #5-73
LN.LOA  = 000004           #5-73
LN.LOO  = 000003           #5-73
LN.OAU  = 000003           #5-73
LN.OFF  = 000001           #5-73
LN.ON   = 000000           #5-73
LN.OOP  = 000004           #5-73
LN.OPE  = 000001           #5-73
LN.REF  = 000002           #5-73
LN.SER  = 000002           #5-73
LN.STA  = 000017           #5-73
LN.SUB  = 000360           #5-73
LN.TRI  = 000006           #5-73
LO.ADD    000022           #5-78
LO.ADR    000052           #5-78
LO.CLS    000002           #5-78
LO.CON    000006           #5-78      *109-4859
LO.COP    000010           #5-78
LO.COS    000020           #5-78
LO.CSR    000010           #5-78       5-78
LO.CTL    000032           #5-78
LO.DES    000002           #5-78       5-78
LO.DTE    000047           #5-78
LO.EVT    000004           #5-78
LO.GRO    000024           #5-78
LO.HOS    000040           #5-78
LO.HTM    000032           #5-78
LO.INA    000062           #5-78
LO.INC    000060           #5-78
LO.LDT    000046           #5-78
LO.LID    000030           #5-78
LO.LIN    000004           #5-78      *109-4858
LO.LTM    000034           #5-78
LO.MAC    000022           #5-78
LO.MDE    000024           #5-78
LO.NAM    000044           #5-78
LO.NET  = 000002           #5-78
LO.NNM    000022           #5-78
LO.OFL    000012           #5-78
LO.ONM    000004           #5-78
LO.OUT    000056           #5-78
LO.OWN    000026           #5-78
LO.PAR    000010           #5-78      *109-4860
LO.PRI    000012           #5-78
LO.RET    000064           #5-78
LO.RPA    000030           #5-78
LO.SCR    000016           #5-78
```

```
                                                                     K  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50       PAGE 20

SYMBOL  CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE                  REFERENCES

.ALLC       004562 R            #24-1487
.ALLOC   =  ******   GX           8-383       11-699      *24-1438    *24-1487
.BASEB   =  140000              #5-75         8-497        15-913
.BUFCT      006424 R            #54-2603
.BUFDF      006324 R            #50-2540
.BUFNM      006356 R            #52-2568
.BUFSK      006342 R            #51-2555
.CRCOV   =  ******   GX           13-824      14-869       24-1353     24-1360
.FLBK0      016372 RG           #103-4493
.FLBK1      016410 RG           #103-4497
.FLBK2      016426 RG           #103-4501
.FLBK3      016444 RG           #103-4505
.FLBK4      016462 RG           #103-4509
.FUDG       004056 R            #24-1378
.HSTAD      007416 R            #63-2838
.HSTAR      007470 R            #63-2854
.MGEXT   =  ******   GX           8-403       11-686       24-1455
.NAM        004464 R              24-1462    #24-1464
.NAM1       004452 R            #24-1461
.NAM2       004460 R            #24-1463
.NODAR      007312 R            #62-2808
.NODNB      007236 R            #62-2791
.NODNM      007056 R            #61-2751
.NOMAP      004426 R            #24-1455
.N1         003662 R            #24-1337
.N2         003672 R            #24-1340
.N3         003710 R            #24-1346
.N4         003726 R            #24-1351
.OCPY       013434 RG           #86-3668
.OEND       013456 RG           #86-3674
.OFLAG      013402 RG           #85-3647
.ONAME      013306 RG           #83-3604
.ONUM       013242 RG           #82-3580
.OVFY       013356 RG           #84-3629
.PLGTH   =  ******   GX           8-267       ?-285       ?-329       17-1043     26-1534     26-1539     26-1569     26-1629
.PNUMB   =  ******   GX           24-1328     24-1331     24-1334     24-1337     24-1343     24-1347     24-1354     24-1361     24-1378
                                  24-1458     24-1487     24-1488     25-1499     25-1502     25-1505     25-1508     52-2571     59-2677
                                  59-2690     60-2725     60-2734     60-2736     62-2794     62-2796     62-2810     62-2813     63-2838
                                  63-2842     63-2854     63-2857     67-3015     67-3018     67-3019     67-3025     67-3028     67-3029
                                  67-3035     67-3038     67-3044     67-3047     67-3054     67-3057     67-3063     67-3066     67-3072
                                  67-3075     67-3081     67-3084     67-3091     67-3094     67-3101     67-3104     67-3111     69-3160
                                  69-3171     69-3206     69-3208     70-3234     70-3241     71-3259     71-3261     72-3282     72-3283
                                  72-3286     73-3304     74-3322     74-3325     74-3328     75-3348     75-3352     75-3355     75-3369
                                  75-3374     76-3389     76-3394     76-3412     76-3415     76-3418     77-3440     77-3442     78-3467
                                  81-3559     81-3561     81-3563     82-3580     82-3586     84-3629     84-3630     85-3647     85-3648
                                  85-3650     86-3668     86-3671     91-3879     91-3882     91-3885     91-3888     91-3894     91-3897
                                  91-3903     91-3912     91-3917
.PNUMH   =  ******   GX           60-2723     73-3300     77-3438     78-3463     82-3584     86-3669     91-3913
.PPRG    =  ******   GX           8-330
.PROMF      006520 R            #56-2638
.PRONM      006500 R            #55-2622
.PSTCN   =  ******   GX           24-1469     68-3134    *68-3139     83-3612     95-4103     95-4105     96-4161    *103-4573
```

```
                .SBTTL  $CXIO - CEX I/O
;+
; $CXIO - CEX I/O (RSX-11M-PLUS ONLY)
;
; THIS ROUTINE IS CALLED TO TRANSFER THE CEX TASK IMAGE INTO MEMORY IN EXEC
; SPACE.
;
; INPUTS:
;       R1=FILE SPEC ADDRESS
;       R2=FILE SPEC LENGTH
;
; OUTPUTS:
;       C-BIT=SUCCESS/FAILURE
;       R0=ERROR CODE
;       R1,R2=DESTROYED
;-

        .IF     DF,R$$MPL

$CXIO:: CLR     ERROR           ; NO ERROR, EITHER
        CALL    $CLOPE          ; OPEN PROCESS FILE
        TSTB    @$CLIOS         ; LOOK AT GCL IOSB
        BMI     101$            ; IF MI, FAILURE
        BIS     (R0)+,(R0)      ; IS FILE CONTIGUOUS?
        BEQ     111$            ; IF EQ, NO
        CALL    PR.LBL          ; READ LABEL BLOCK
        BCS     10$             ; IF CS, ERROR
        CALL    PR.XFR          ; READ PROCESS TASK IMAGE INTO CORE
10$:    CALL    $CLDEQ          ; CLOSE PROCESS FILE
        TST     ERROR           ; ANY ERRORS? (CLEARS C-BIT)
        BEQ     30$             ; IF ZERO, NO
20$:    SEC                     ; INDICATE FAILURE
30$:    MOV     ERROR,R0        ; GET ERROR CODE
        RETURN

;
; ERROR CONDITIONS
;
101$:   MOV     #ERR2,ERROR     ; OPEN FAILURE
        BR      20$
111$:   MOV     #ERR6,ERROR     ; NOT CONTIGUOUS
        BR      10$

        .ENDC
```

```
                                                 .SBTTL  PR.SUB - CREATE SUB-PARTITION
                                         ;+
                                         ; PR.SUB - CREATE SUB-PARTITION
                                         ;
                                         ; THIS ROUTINE IS CALLED IF THE PROCESS PARTITION IS SYSTEM CONTROLLED.
                                         ; A NEW SUB-PARTITION IS CREATED AND LINKED IN. THE PRESENCE OF AN MCB
                                         ; PROCESS IN A SYSTEM CONTROLLED PARTITION IS DENOTED BY SETTING THE COMMON
                                         ; BIT (PS.COM) IN THE PCB STATUS WORD (P.STAT) AND BY SETTING THE PARTITION
                                         ; NAME (P.NAM) TO "NT.***", WHERE "***" IS THE THREE CHARACTER RAD50 PDV NAME.
                                         ;
                                         ; INPUTS:
                                         ;       R5=MAIN PARTITION PCB ADDRESS
                                         ;
                                         ; OUTPUTS:
                                         ;       R0,R1,R2,R3,R4=DESTROYED
                                         ;       R5=SUB-PARTITION PCB ADDRESS
                                         ;-

002334  016701  000000G                  PR.SUB: MOV     .PLGTH,R1       ; GET PCB LENGTH
002340                                           CALL    $ALL15          ; TRY TO ALLOCATE SUB-PARTITION PCB
002344  103002                                   BCC     10$             ; IF C CLEAR, THEN OK
002346  000167  000426                           JMP     100$            ; FAILURE THEN
002352  012701  000000G                  10$:    MOV     #$BFR,R1        ; COPY PCB ADDRESS
002356  016702  000000G                          MOV     .PLGTH,R2       ; GET PCB LENGTH
002362  006202                                   ASR     R2              ; CONVERT TO A WORD COUNT
002364  005021                           20$:    CLR     (R1)+           ; ZERO THE ENTIRE SUB-PCB
002366  005302                                   DEC     R2
002370  001375                                   BNE     20$             ; ...

002372  032767  000001  000150'                  BIT     #MCLOA,MCFLG    ; MICROCODE PARTITION ?
002400  001407                                   BEQ     30$             ; IF NO - BRANCH
002402  016767  000000G 000004G                  MOV     $NAME,P.NAM+$BFR ; LOOK FOR PCB
002410  012767  050670  000006G                  MOV     #^RMC,P.NAM+2+$BFR ; ...
002416  000406                                   BR      40$             ; CONTINUE

002420  012767  055274  000004G  30$:    MOV     #^RNT.,P.NAM+$BFR ; SET NAME TO "NT.***"
002426  016767  000000G 000006G          MOV     $NAME,P.NAM+2+$BFR ;    ...

002434  010567  000012G          40$:    MOV     R5,P.MAIN+$BFR  ; MAIN PCB ADDRESS
002440  016767  000142' 000016G          MOV     BLOX,P.BLKS+$BFR        ; SIZE IN 32. WORD BLOCKS

                                         .IF DF  R$$MPL                                                  ;[MP01]

                                         MOV     #PS.COM!PS.NSF!PS.FXD!PS.CHK,P.STAT+$BFR ; SET STATUS   ;[MP01]
                                         MOV     #P2.RON,P.ST2+$BFR      ; READ ONLY                     ;[MP01]

                                         .IFF                                                            ;[MP01]
002446  012767  000640  000030G          MOV     #PS.COM!PS.NSF!PS.SYS,P.STAT+$BFR ; COMMON, NO SHUFFLING;[MP01]

                                         .ENDC                                                           ;[MP01]

002454                                   PUTRC   R0,.PLGTH       ; WRITE INITIALIZED PCB TO SYSTEM IMAGE
002472  010501                           MOV     R5,R1           ; COPY MAIN PCB ADDRESS
002474  005002                           CLR     R2              ; NO PREVIOUS SUB-PCB
002476  005046                           CLR     -(SP)           ; PREVIOUS SUB-PCB IF SPACE IS FOUND
002500                                   GETRV   R1              ; READ IN PCB
002510  066702  000014G          50$:    ADD     P.REL+$BFR,R2   ; GIVING BIAS OF NEXT HOLE IN PARTITION
```

Table of contents

```
     5-   60    MACROS AND SYMBOLS AND INITIALIZATION
     6-   75    PROMPTS
```

```
                                          .SBTTL  LOCAL DATA


                                  ;***
                                  ; LOCAL DATA
                                  ;***

000000                                    .PSECT  DATA,D

                                          .NLIST  BEX

000000      044     130     130   VECTAB: .ASCII  "$XXX"
000004      040                           .BYTE   40
000005      040                           .BYTE   40
000006      044     130     130   VECLEN: .ASCII  "$XXX"
000012      040                           .BYTE   40
000013      040                           .BYTE   40

                                          .IF DF  R$$MPL                                          ;[MP01]

                                  ;
                                  ; PROCESS VECTOR TABLE SYMBOLS
                                  ;
                                  CETNAM: .ASCIZ  /CETAB/          ; CETAB FILE NAME              ;[MP01]

                                  CTBLK:  .WORD   0,0              ; CONTROLLER SYMBOL NAME       ;[MP01]
                                  CTBSYM: .RAD50  "$CTXXX"                                        ;[MP01]
                                  CTVAL:  .BLKW   1                                               ;[MP01]
                                          .ENDC                                                   ;[MP01]


000014                            XPTPDV: .BLKW   1                ; PDV ADDRESS FOR XPT
000016                            XPTDSP: .BLKW   1                ; XPT'S ADDRESSING BIAS
                                  ;
                                  ; SYMBOL TABLE EXTENSION
                                  ;
000020      123     124     102   STBEXT: .ASCIZ  "STB"

                                  ;
                                  ; ASCII DISPATCH TABLE NAME
                                  ;
000024      044     130     130   DSPTAB: .ASCII  "$XXXTB"

                                          .EVEN

                                  ;
                                  ; PROCESS DISPATCH TABLE SYMBOL
                                  ;

                                          .IF DF  R$$MPL

                                  PBLK:   .WORD   VECSYM,0
                                          .IFF
000032   000070' 000000           PBLK:   .WORD   $PRINC,0
                                          .ENDC
000036                            PSYM:   .BLKW   2
```

```

                                          .SBTTL  $CACH - ALLOCATE CACHE
                                  ;+
                                  ; $CACH - ALLOCATE CACHE
                                  ;
                                  ; INPUTS:
                                  ;         $PDVA - XPT PDV ADDRESS
                                  ;
                                  ;
                                  ; OUTPUTS:
                                  ;         CACHE IS ALLOCATED IN XPT PROCESS SPACE FOR ENDNODES
                                  ;
                                  ;                 N$CACH - SIZE OF CACHE
                                  ;
                                  ;-

002000                            $CACH:: GETRV   $DECPT,#D$END           ; READ DECNET HOME BLOCK
002020  005767  000060G                   TST     $BFR+D$NN               ; ENDNODE ?
002024  001155                            BNE     110$                    ; IF NOT - BRANCH

002026  005000                    10$:    CLR     R0                      ; INIT REG
002030  156700  000040G                   BISB    $BFR+D$MLL,R0           ; READ MAX LOGICAL LINKS VALUE
002034  010002                            MOV     R0,R2                   ; SAVE NUMBER OF CACHE ENTRIES
002036  012701  000012                    MOV     #C$LEN,R1               ; GET CACHE ENTRY SIZE
002042                                    CALL    $MUL                    ; CALCULATE TOTAL SIZE

002046  016700  000000G                   MOV     $PDVA,R0                ; GET XPT PDV ADDRESS
002052  016067  000000G 000016'           MOV     Z.DSP(R0),XPTDSP        ; GET XPT BIAS
002060  016000  000000G                   MOV     Z.DAT(R0),R0            ; GET XPT DDB ADDRESS
002064                                    GETRV   R0,#N$XLEN              ; READ XPT'S DDB

002102  010267  000062G                   MOV     R2,$BFR+N$CACH          ; STORE NUMBER OF CACHE ENTRIES
002106  016702  000016'                   MOV     XPTDSP,R2               ; SAVE XPT BIAS
002112  016700  000064G                   MOV     $BFR+N$CACH+2,R0        ; GET ADDRESS OF NI CACHE
002116                                    PUTRC                           ; WRITE UPDATED XPT DDB TO IMAGE
002122  022700  140000                    CMP     #140000,R0              ; IS CACHE IN SECOND 4KW BLOCK ?
002126  003003                            BGT     20$                     ; IF NO - BRANCH
002130  062702  000200                    ADD     #200,R2                 ; ADJUST RELOCATION BIAS TO NEXT 4KW
002134  000454                            BR      70$                     ; CONTINUE

002136  010103                    20$:    MOV     R1,R3                   ; COPY SIZE
002140  060003                            ADD     R0,R3                   ; CALCULATE TOP OF NI CACHE
002142  022703  140000                    CMP     #140000,R3              ; IS TOP INTO SECOND 4KW BLOCK ?
002146  002047                            BGE     70$                     ; IF NO -  BRANCH

002150  162703  140000                    SUB     #140000,R3              ; CALCULATE SIZE IN SECOND BLOCK
002154  160301                            SUB     R3,R1                   ; CALCULATE SIZE IN FIRST BLOCK
002156  010105                            MOV     R1,R5                   ; COPY SIZE
002160  016702  000016'                   MOV     XPTDSP,R2               ; COPY XPT PROCESS BIAS

002164  022705  001000            30$:    CMP     #512.,R5                ; MORE THAN ONE TRANSFER NEEDED ?
002170  002002                            BGE     40$                     ; IF NO - BRANCH
002172  012701  001000                    MOV     #512.,R1                ; LIMIT TO MAXIMUM SIZE (1000 BYTES)

002176  012704  000000G           40$:    MOV     #$BFR,R4                ; GET BUFFER ADDRESS
002202  010102                            MOV     R1,R2                   ; COPY SIZE
002204  105024                    50$:    CLRB    (R4)+                   ; INIT BUFFER
002206                                    SOB     R2,50$                  ; IF MORE - BRANCH
```

```
                                                        K  8
VPRD - VNP SET PROCESS ROUTINES MACRO V05.03b Wednesday 28-Aug-85 20:04  Page 19-2
Symbol table

ALVBLK  000346R       D$NBRA  000054        LF.MFL= 004000        LS.PWF= 040000        N$GTM   000015
AT$ACL= 000100        D$NEND= 000054        LF.MTP= 000020        LS.TOP= 002000        N$HC1   000052
AT$ADN= 000040        D$NLN   000030        LF.PAC= 000200        LS.UNF= 020000        N$HC2   000056
AT$AUP= 000020        D$NN    000060        LF.RDY= 040000        LS.X25= 000012        N$HIGH  000033
AT$CYC= 000004        D$OUTT  000043        LF.REA= 010000        LS.11D= 004000        N$LLT   000026
AT$LV1= 000002        D$RETF  000050        LF.SER= 000040        LX.CEX= 000004        N$LLTM  000024
AT$LV2= 000001        D$RNN   000002        LF.TIM= 000010        LX.LIN= 000006        N$LVC   000036
AT$NEX= 000200        D$RTMR  000076        LF.UNL= 020000        LX.PRO= 000005        N$LV1   000002
AT$UP = 000010        D$SEG   000036        LF.X2P= 000000        L$$ASG= 000000        N$LV2   000010
AXSYM   000460R       D$SER   000032        LN.CLO= 000000        L$$DRV= 000000        N$MBXQ  000050
A$CIR   000003        D$SQRL  000052        LN.DUM= 000005        L$$P11= 000001        N$MHC1  000036
A$LEN   000004        D$$BUG= 177514        LN.LOA= 000004        L$$11R= 000000        N$MHC2  000044
A$NID   000000        D$$ISK= 000000        LN.LOD= 000003        L.COST  000015        N$PLD   000016
A$TM    000003        D$$L11= 000001        LN.OAU= 000003        L.CTL   000012        N$PLLT  000030
A$TMI   000002        D$$YNC= 000000        LN.OFF= 000001        L.CVA   177776        N$PRI   000076
A$TSZ   000000        D$$YNM= 000000        LN.ON = 000000        L.DDM   000002        N$RDA1  000022
A$TYP   000002        ETAB1   000222R   002 LN.DDP= 000004        L.DDS   000004        N$RDA2  000030
A$$CHK= 000000        ETAB2   000232R   002 LN.OPE= 000001        L.DLC   000003        N$RTMX  000014
A$$CPS= 000000        E$NBR   000014        LN.REF= 000002        L.DLM   000006        N$RTM1  000014
A$$PRI= 000000        E$NBS   000020        LN.SER= 000002        L.DLS   000010        N$RTM2  000015
A$$TRP= 000000        E$NCR   000034        LN.STA= 000017        L.FLG   000000        N$RT1   000000
C$LEN   000012        E$NCS   000036        LN.SUB= 000360        L.KRBA  000016        N$RT2   000006
C$NID   000000        E$NIC   000044        LN.TRI= 000006        L.LEN = 000022        N$SLA   000016
C$NXH   000002        E$NLEN  000050        LO.CON  000006        L.MPF   000022        N$SNDD  000012
C$TIM   000010        E$NLLA  000012        LO.CSR  000006        L.NMST  000020        N$TCTL  000112
C$$CKP= 000000        E$NLNK  000000        LO.CTL  000014        L.NSTA  000014        N$TIM   000004
C$$DRE= 000400        E$NML   000040        LO.LIN  000004        L.OWNR  000021        N$TLC   000100
C$$RSH= 177564        E$NMR   000024        LO.PAR  000010        L.UNT   000013        N$TNC   000114
DSPTAB  000024R   002 E$NMS   000030        LO.PRI  000010        MILBLK  000140R   002 N$TRC   000106
DTBLK   000114R   002 E$NNDD  000002        LO.SCR  000014        M$$CRB= 000124        N$TTCB  000110
DTSYM   000120R   002 E$NRT   000042        LO.STA  000015        M$$CRX= 000000        N$VCB   000010
DTVAL   000124R   002 E$NRTP  000005        LO.VCT  000012        M$$FCS= 000000        N$VER   000066
D$AMXC  000072        E$NSEG  000010        LR.AST  000026        M$$MGE= 000000        N$XLEN  000124
D$AMXH  000074        E$NTIM  000046        LR.CIR  000002        M$$NET= 000000        N$$ACC= 000001
D$ANN   000000        E$NUSE  000004        LR.CTL  000004        M$$OVR= 000000        N$$BUF= 000001
D$BRPR  000102        E$STRT  000006        LR.EFN  000004        NF$BLK= 000100        N$$LDV= 000001
D$BRTM  000100        E$$XPR= 000000        LR.LIN  000002        NF$DMO= 000010        N$$MCP= 000001
D$DELF  000045        FNDPDV  004426R       LR.PRM  000030        NF$MOU= 000040        N$$MLL= 000001
D$DELW  000046        F$$LVL= 000001        LR.PRO  000002        NF$RST= 000002        N$$MOV= 000010
D$END = 000104        F.RSIZ= ****** GX     LR.STS  000000        NF$SCN= 000020        N$$NCT= 000001
D$FNB   000034        G$$TPP= 000000        LR.TCB  000002        NF$SHU= 000004        N$$PEM= 000001
D$HIDR  000024        G$$TSS= 000000        LR.UNT  000005        NF$TIM= 000200        PBLK    000032R   002
D$HOST  000022        G$$TTK= 000000        LS.AST= 100000        N$ACQ   000000        PRSYM   000600R
D$INAC  000044        G$$WRD= 000000        LS.CEX= 000001        N$ACTL  000032        PR1SC   000162R   002
D$INCT  000042        I$$RAR= 000000        LS.CIR= 000010        N$ADJ1  000072        PR100M  000150R   002
D$IPL   000051        I$$RDN= 000000        LS.CXO= 010000        N$ADJ2  000074        PSYM    000036R   002
D$LID   000020        K$$CNT= 177546        LS.CX5= 000400        N$CACH  000062        PVAL    000042R   002
D$LNAM  000006        K$$CSR= 177546        LS.DLM= 004000        N$CIR   000034        P$$P45= 000000
D$LNUM  000014        K$$LDC= 000000        LS.ECH= 001000        N$CRC   000120        P$$WRD= 000000
D$LST   000047        K$$TPS= 000074        LS.FDX= 004000        N$DLA   000020        P.BLKS= ****** GX
D$MAXC  000064        LD$LP = 000000        LS.HDX= 010000        N$DLY   000014        P.REL = ****** GX
D$MAXH  000066        LF.ACT= 100000        LS.LIN= 000003        N$ELEN  000054        Q$$OPT= 000010
D$MAXV  000070        LF.BRO= 000400        LS.LMC= 000007        N$ENC   000042        R$$DER= 000000
D$MLL   000040        LF.BWT= 000007        LS.NTI= 004000        N$ERRC  000022        R$$K11= 000001
D$MNDD  000041        LF.ENA= 002000        LS.ON = 000011        N$FLG   000005        R$$SND= 000000
D$NA    000062        LF.LPB= 001000        LS.OPT= 000400        N$FNC   000006        R$$11M= 000000
D$NBEA  000056        LF.MDC= 000100        LS.PRO= 000002        N$GENQ  000052        SECBLK  000152R   002
```

```
                .IF DF M$$PRO                                           ;[MP01]
                .IIF EQ M$$PRO - 1 .TITLE VRES1                         ;[MP01]
                .IIF EQ M$$PRO - 2 .TITLE VRES2                         ;[MP01]
                .IIF EQ M$$PRO - 3 .TITLE VRES3                         ;[MP01]
                .IIF EQ M$$PRO - 4 .TITLE VRES4                         ;[MP01]
                .IFF                                                    ;[MP01]
                .TITLE  VRES - VNP ROOT

                .ENDC                                                   ;[MP01]

                .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       VNP - ROOT SEGMENT DATA
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
```

```
    402                                         .SBTTL  ERROR HANDLING
    403                                 ;
    404                                 ; ERROR REPORTING
    405                                 ;
    406
    407                176031           FATAL==-999.                    ; FATAL ERRORS HAVE R2 SET TO THIS VALUE
    408
    409 001622     126     116     120  PREFIX: .ASCII  /VNP -- /       ; PREFIX MESSAGE
    410 001631     052     106     101  PREF2:  .ASCIZ  /*FATAL* /      ; FATAL ERROR MESSAGE NOTIFICATION
    411                                         .EVEN
    412
    413 001642                          $FNOUT::
    414 001642 012700  001364'                  MOV     #FNERR,R0       ; GET MESSAGE ADDRESS
    415 001646 016701  177510                   MOV     FNPTR,R1        ; CALCULATE LENGTH
    416 001652 160001                           SUB     R0,R1           ; ...
    417 001654                          $TIOUT::
    418 001654 010067  176134                   MOV     R0,TIOUT+Q.IOPL         ; BUFFER ADDRESS
    419 001660 010167  176132                   MOV     R1,TIOUT+Q.IOPL+2       ;  AND LENGTH
    420 001664                                  DIR$    #TIOUT                  ; WRITE IT TO TI:
    421 001672                                  RETURN
    422
    423 001674                          $EROUT::
    424 001674 105067  177731                   CLRB    PREF2           ; ASSUME NON-FATAL ERROR ...
    425 001700 022701  176031                   CMP     #-999.,R1       ; IS THIS A FATAL ERROR ?
    426 001704 001003                           BNE     5$              ; NO
    427 001706 112767  000052  177715           MOVB    #'*,PREF2       ; INCLUDE '*FATAL*' MESSAGE
    428 001714 010146                   5$:     MOV     R1,-(SP)                ; SAVE POSSIBLE FLAG
    429 001716 010046                           MOV     R0,-(SP)                ; PRESERVE R0
    430
    431 001720 016700  000310'                  MOV     $CLBUF,R0               ; SET ADDRESS OF OUTPUT BUFFER
    432 001724 012701  001622'                  MOV     #PREFIX,R1              ; SET ADDRESS OF PREFIX
    433 001730 112120                   10$:    MOVB    (R1)+,(R0)+             ; MOVE PREFIX INTO PLACE
    434 001732 001376                           BNE     10$                     ; ...
    435 001734 005300                           DEC     R0                      ; SET ADDRESS OF OUTPUT BUFFER
    436 001736                                  CALL    $ERMSG                  ; REFORMAT ERROR MESSAGE
    437 001742 060001                           ADD     R0,R1                   ; FORM ADDRESS OF TERMINATOR
    438 001744 016700  000310'                  MOV     $CLBUF,R0               ; SET ADDRESS OF OUTPUT STRING
    439 001750 160001                           SUB     R0,R1                   ; FORM TOTAL STRING LENGTH
    440 001752                                  CALL    $TIOUT                  ; OUTPUT MESSAGE
    441
    442 001756 012600                           MOV     (SP)+,R0                ; RESTORE R0
    443 001760 012601                           MOV     (SP)+,R1                ; AND R1
    444 001762 022701  176031                   CMP     #-999.,R1               ; IS THIS FATAL ERROR
    445 001766 001002                           BNE     20$                     ; NO
    446 001770 000167  000000G                  JMP     $PANIC                  ; YES
    447 001774 005726                   20$:    TST     (SP)+                   ; DON'T RETURN HERE
    448 001776 000261                           SEC                             ; WARNINGS SET C-BIT
    449 002000                                  RETURN                          ; BUT HERE
```

```
                                          .SBTTL  $PUTRC - PUT RECORD TO SYSTEM IMAGE FILE
                                  ;+
                                  ;       **-$PUTRC
                                  ;
                                  ;  THIS ROUTINE IS CALLED TO PUT A RECORD INTO THE TASK IMAGE FILE. THE
                                  ;  ADDRESS OF THE RECORD IS DEFINED BY THE CONTENTS OF $RADDR AND THE SIZE
                                  ;  BY THE SYSTEM IMAGE FILE FDB.
                                  ;
                                  ;  INPUTS:
                                  ;       $RADDR          -ADDRESS OF THE RECORD.
                                  ;       SYSFDB+F.RSIZ   -SIZE OF THE RECORD.
                                  ;       $BFR            -RECORD TO BE WRITTEN.
                                  ;
                                  ;  OUTPUTS:
                                  ;       THE RECORD CONTAINED IN $BFR IS WRITTEN INTO THE TASK IMAGE AT
                                  ;       THE ADDRESS SPECIFIED BY $RADDR.
                                  ;
                                  ; NOTE: ALL REGISTERS ARE PRESERVED ACROSS CALL.
                                  ;-

 004444                           $PUTRC::
                                          .IF DF  R$$MPL                                                  ;[MP01]

                                          BIT     #160000,$RADDR  ; TRYING TO ACCESS FIRST 4KW ?
                                          BEQ     5$              ; IF YES - BRANCH

                                          BIT     #DF.DAS,$DFLAG  ; D-SPACE ENABLED ?                     ;[MP01]
                                          BEQ     5$              ; IF NO - BRANCH                        ;[MP01]
                                          MOV     $RBLCK,RB       ; SAVE BIAS                             ;[MP01]
                                          MOV     .DBIAS,$RBLCK   ; MAP TO RSX POOL                       ;[MP01]
                                          MOV     $RADDR,RA       ; SAVE VIRTUAL ADDRESS                  ;[MP01]

                                          CALL    DBLK            ; CALCULATE BLOCK AND ADDRESS           ;[MP01]
                                          CALL    $PUTAD          ; DO A BIASED WRITE                     ;[MP01]
                                          MOV     RA,$RADDR       ; RESTORE VIRTUAL ADDRESS               ;[MP01]
                                          MOV     RB,$RBLCK       ; RESTORE BIAS
                                          RETURN

                                  5$:
                                          .ENDC                                                           ;[MP01]

 004444  004267  000000G          $PUTRD::JSR     R2,$SAVVR       ; SAVE NONVOLATILE REGISTERS
 004450  012746  000000G                  MOV     #.PUT,-(SP)     ; SET ADDRESS OF RECORD ACCESS ROUTINE
 004454  010346                   20$:    MOV     R3,-(SP)        ; SAVE R3
 004456  012700  000256'                  MOV     #SYSFDB,R0      ; GET ADDRESS OF INPUT FILE FDB
 004462  005001                   22$:    CLR     R1              ; CLEAR HIGH PART OF VIRTUAL BLOCK NUMBER
 004464  016702  173766                   MOV     $RADDR,R2       ; GET ADDRESS OF RECORD IN TASK IMAGE
                                          .IF DF  R$$MPL                                                  ;[MP01]
                                          BIT     #DF.SCO,$DFLAG  ; $SCVOL READ ?
                                          BNE     25$             ; IF YES - BRANCH
                                          .ENDC                                                           ;[MP01]
 004470  005767  173772                   TST     $XFRAD          ; IS THIS A VIRGIN SYSTEM?
 004474  001021                           BNE     25$             ; IF NE YES
 004476  020267  173770                   CMP     R2,$LOWSZ       ; REFERENCING DISPLACED MEMORY ?
 004502  103016                           BHIS    25$             ; IF HIS NO
 004504  016702  173752                   MOV     $VCTVB,R2       ; GET VBN OF MEMORY DISPLACED BY 'SAV'
 004510  016703  173750                   MOV     $VCTOF,R3       ; GET BLOCK OFFSET OF LOCN 0
```

```
VRES - VNP ROOT MACRO V05.03b Tuesday 17-Sep-85 16:15  Page 26-4
Symbol table

$TIOUT  001654RG        $X2HLD  003404RG        $$LRB   003424RG        .CRAVL  003552RG        .PARHD  003564RG
$TK100  004114RG        $X2MBS  003376RG        $$LSAD  003372RG        .CRCOV  003666RG        .PATT   003646RG
$TMPBF  000000RG    004 $X2MRT  003400RG        $$LTIM  002176RG        .DEUMR  003634RG        .PDVTA  003660RG
$TMPND  000000RG    005 $X2MWS  003374RG        $$MTP   002512RG        .DEVHD  003556RG        .PHDR   003650RG
$TRIB   002340RG        $X2RET  003402RG        $$MUX   002662RG        .DLCOV  003664RG        .PLGTH  003602RG
$TYPE   002012RG        $X3CAL  002376RG        $$NBRA  002656RG        .EXSIZ  003560RG        .POINT= ****** GX
$T1SCL  004012RG        $X3CLR  002400RG        $$NODC  002336RG        .FMASK  003562RG        .PPRO   003652RG
$T100C  003772RG        $X3DBS  002366RG        $$PAR   002352RG        .FRKHD  003574RG        .PUT   = ****** GX
$T100Q  003764RG        $X3DWS  002372RG        $$PCHA  003230RG        .FSRCB= ****** G        .RDBNM  001014RG
$T50CL  003772RG        $X3MBS  002370RG        $$PRI   002506RG        .GET   = ****** GX      .RDBSZ  001016RG
$T50Q   003764RG        $X3MCL  002406RG        $$RPRI  002660RG        .HEADR  003576RG        .ROUTM  002032RG
$UNBAD  000510RG        $X3MRE  002410RG        $$SCSR  002664RG        .MAXC   002026RG        .SDBNM  001020RG
$UNBLK  000512RG        $X3MRS  002412RG        $$SNUM  002514RG        .MAXH   002030RG        .SDBSZ  001022RG
$UNBSZ  000506RG        $X3MWS  002374RG        $$TBL   002360RG        .MGEXT  003626RG        .STKDP  003600RG
$UNEND  000612RG        $X3RES  002402RG        $$VECT  002510RG        .MGMGE  003630RG        .SYSIZ  003566RG
$VACP  = ****** GX      $X3RST  002404RG        $$$    = 005450R        .NETPF  003654RG        .SYSNM  003570RG
$VCTOF  000464RG        $ZTIME  003746RG        $$$ARG= 000003          .NN     002024RG        .TLGTH  003642RG
$VCTVB  000462RG        $$     = 177777         $$$OST= 000010          .NS0    003606RG        .TSKHD  003636RG
$VFLAG  003412RG        $$APR   002574RG        $$$T1  = 000002         .NS1    003610RG        .UMRHD  003632RG
$VFLG1  003414RG        $$CSR   002504RG        $$$T2  = 000166R    003 .NS2    003612RG        .VCTR   003604RG
$VGCL   005022RG        $$DCHA  002724RG        $$$$   = 005616R        .NS3    003614RG        ..C00   003554RG
$XAVL   004106RG        $$DEV   002356RG        .ABTIM  003640RG        .NS4    003616RG        ..SCSR  002034RG
$XBIAS  004104RG        $$DPR   002654RG        .ALLOC  001012RG        .NS5    003620RG        ...PC1= 000256R
$XBUF2  003420RG        $$EXT   002022RG        .CEAVL  003662RG        .NS6    003622RG        ...PC2= 000452R
$XFRAD  000466RG        $$HTIM  002036RG        .CEPWR  003670RG        .NS7    003624RG        ...PC3= 000256R
$XFRSV  000470RG        $$INC   002350RG        .CLINS  003572RG        .NTUIC  003656RG        ...TPC= 000006
$XLEN2  003422RG        $$LINX  002362RG        .CLKHD  003644RG

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        005756    001   (RW,I,LCL,REL,CON)
$$FSR1  002040    002   (RW,D,GBL,REL,OVR)
$$$GCL  000502    003   (RW,I,LCL,REL,CON)
TMP$$A  001200    004   (RW,I,LCL,REL,OVR)
TMP$$B  000000    005   (RW,I,LCL,REL,CON)
CET$$A  000000    006   (RW,I,LCL,REL,OVR)
CET$$B  000000    007   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 107
Work  file writes: 90
Size of work file: 22203 Words  ( 87 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:01:20.32
DB2:VRES.T50,[132,134]VRES/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VRES
```

```
VRES       CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 13
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE            REFERENCES

$XLEN2    003422 RG      #15-630
$X2HLD    003404 RG      #15-563
$X2MBS    003376 RG      #15-560
$X2MRT    003400 RG      #15-561
$X2MWS    003374 RG      #15-559
$X2RET    003402 RG      #15-562
$X3CAL    002376 RG      #15-528
$X3CLR    002400 RG      #15-529
$X3DBS    002366 RG      #15-524
$X3DWS    002372 RG      #15-526
$X3MBS    002370 RG      #15-525
$X3MCL    002406 RG      #15-532
$X3MRE    002410 RG      #15-533
$X3MRS    002412 RG      #15-534
$X3MWS    002374 RG      #15-527
$X3RES    002402 RG      #15-530
$X3RST    002404 RG      #15-531
$ZTIME    003746 RG      #17-778
$$      = 177777          23-1282    23-1282   #23-1282    23-1283    23-1283   #23-1283    23-1284    23-1284   #23-1284
                         #23-1285    23-1286    23-1286   #23-1286   #23-1287    23-1288    23-1288   #23-1288   #23-1289
                          23-1290    23-1290   #23-1290    23-1291    23-1291   #23-1291   #23-1292
$$APR     002574 RG      #15-548
$$CSR     002504 RG      #15-543
$$DCHA    002724 RG      #15-554
$$DEV     002356 RG      #15-520
$$DPR     002654 RG      #15-549
$$EXT     002022 RG      #15-499
$$HTIM    002036 RG      #15-510
$$INC     002350 RG      #15-518
$$LINX    002362 RG      #15-522
$$LRB     003424 RG      #15-634
$$LSAD    003372 RG      #15-558
$$LTIM    002176 RG      #15-511
$$MTP     002512 RG      #15-546
$$MUX     002662 RG      #15-552
$$NBRA    002656 RG      #15-550
$$NODC    002336 RG      #15-512
$$PAR     002352 RG      #15-519
$$PCHA    003230 RG      #15-556
$$PRI     002506 RG      #15-544
$$RPRI    002660 RG      #15-551
$$SCSR    002664 RG      #15-553
$$SNUM    002514 RG      #15-547
$$TBL     002360 RG      #15-521
$$VECT    002510 RG      #15-545
$$$     = 005450 R       #23-1282    23-1285   #23-1286    23-1287   #23-1288    23-1289   #23-1290    23-1292
$$$ARG  = 000003         #6-96       6-96      #6-96       6-96      #6-96       6-96      #6-96       6-96       6-96
                         #6-102      6-102     #6-102      6-102     #6-102      6-102     #6-102      6-102      6-102
$$$GLB  = ******          6-96       6-102      6-102      6-102      6-102      6-102
$$$T1   = 000002         #6-102
$$$T2   = 000166 R       #6-102      6-102
$$$$    = 005616 R       #23-1285   #23-1287   #23-1289   #23-1292
```

```
    296                                               .SBTTL  NOP - NO PRELIMINARY WORK REQUIRED
    297                                       ;+
    298                                       ;       NOP     - NO CHECKING OR LOOKUPS NEEDED
    299                                       ;-
    300
    301 000120  000241                        NOP:    CLC
    302 000122                                        RETURN
    702 001456  010067  000000G          60$:     MOV     R0,..SCSR        ; SAVE SECONDARY CSR
    703 001462  012767  000001  000000G           MOV     #1,$$MTP         ; DEFAULT TO ONE STATION
    704 001470  112767  000001  000000G           MOVB    #1,$$SNUM        ; AND STATION NUMBER 1 (POINT-TO-POINT)
    705 001476  000241                            CLC                      ; INDICATE SUCCESS
    706 001500                                    RETURN
    707
    708                                  ;
    709                                  ; ERROR CONDITIONS
    710                                  ;
    711 001502                           101$:    EMSG$R  94               ; ILLEGAL CSR VALUE
    712 001510                           111$:    EMSG$R  93               ; ILLEGAL PRIORITY VALUE
    713 001516                           121$:    EMSG$R  92               ; ILLEGAL VECTOR VALUE
    714 001524                           131$:    EMSG$R  ZP               ; VECTOR NOT IN SYSTEM
    1123 003242                                 EMSG$R  87                ; UNITS OUT OF ORDER
```

```
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE             REFERENCES

LO.SEG    000054          #5-78
LO.SKA    000014          #5-78
LO.SKN    000014          #5-78
LO.SNM    000035          #5-78
LO.TPA    000020          #5-78
LO.TRB    000034          #5-78
LO.TRI    000030          #5-78
LO.UNT    000033          #5-78
LO.VCT    000014          #5-78
LO.VER    000042          #5-78
LR.ACC    000020          #5-78
LR.ACO    000052          #5-78
LR.ADD    000012          #5-78
LR.ALI    000002          #5-78
LR.BLK    000010          #5-78
LR.CTL    000004          #5-78    *107-4785  *108-4823
LR.DES    000010          #5-78
LR.HLP    000002          #5-78
LR.LIN    000002          #5-78     5-78       5-78       5-78       5-78       5-78       5-78      *107-4788  *108-4826
LR.NAM    000012          #5-78
LR.PAS    000041          #5-78
LR.PRO    000002          #5-78    *109-4857
LR.STS    000000          #5-78    *107-4777  *107-4784  *108-4815  *108-4822  *109-4849  *109-4856
LR.TRI    000006          #5-78    *107-4787  *108-4825
LR.TYP    000002          #5-78
LR.UCB    000016          #5-78
LR.UIC    000020          #5-78
LR.UNT    000005          #5-78    *107-4786  *108-4824
LST       011074 R        #67-3091
LS.ALI  = 000016          #5-78
LS.CEX  = 000001          #5-78     8-366
LS.CIR  = 000033          #5-78     108-4815
LS.CXO  = 010000          #5-78
LS.CX5  = 000400          #5-78
LS.ECH  = 001000          #5-78
LS.FDX  = 004000          #5-78
LS.HDX  = 010000          #5-78
LS.HLP  = 000012          #5-78
LS.LIN  = 000003          #5-78     107-4777
LS.LMC  = 000007          #5-78
LS.LOG  = 000020          #5-78
LS.MOD  = 000036          #5-78
LS.NOD  = 000014          #5-78
LS.NTI  = 000013          #5-78
LS.OBJ  = 000022          #5-78
LS.OPT  = 000400          #5-78
LS.PRO  = 000002          #5-78     109-4849
LS.TOP  = 002000          #5-78     107-4784   108-4822   109-4856
LS.TST  = 000010          #5-78
LS.UNF  = 020000          #5-78
LS.XIT  = 000011          #5-78
LX.ALI  = 000017          #5-78
```

```
                                                               L  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 21
SYMBOL  CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE                REFERENCES

.PSTPT   = ******   GX         24-1468    55-2622    61-2756    64-2883    65-2909    68-3137    83-3604    91-3906    93-3989
                               95-4106    96-4160    103-4569
.RADDR     011674 R           #70-3233
.RAMXC     012412 R           #76-3389
.RAMXH     012340 R           #75-3369
.RARE      011614 R           #69-3206
.RBEA      012250 R           #75-3343
.RBTM      012464 R           #76-3407
.RCOST     011746 R           #71-3256
.RDBNM   = ******   GX        *24-1340    24-1351
.RDBSZ   = ******   GX        *24-1343    24-1352
.RHOP      012022 R           #72-3279
.RMAXA     012160 R           #74-3318
.RNAME     011254 R           #68-3130
.RNUM      011330 R           #69-3160
.RRTM      012110 R           #73-3300
.R1        003632 R           #24-1328
.R2        003642 R           #24-1331
.R3        003652 R           #24-1334
.SBPCB   = ******   GX        *8-476     *8-486
.SBRA      007030 R           #60-2734
.SCTIM     007002 R           #60-2722
.SCTL      006604 R           #59-2677
.SDBNM   = ******   GX        *24-1346    24-1358
.SDBSZ   = ******   GX        *24-1347    24-1359
.SLTDF     006562 R           #57-2659
.STADF     006576 R           #58-2670
.STRNG     007706 R           #65-2908
.SUNT      006614 R           #59-2683
.SYSID     007574 R           #64-2883
.SYSNM   = ******   GX         61-2778
.TPARS   = ******   GX         30-1846    31-1875
```

```
                                                               M  2
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 22
```

```
                                                .SBTTL  $PRIO - PROCESS I/O
                                        ;+
                                        ; $PRIO - PROCESS I/O
                                        ;
                                        ; THIS ROUTINE IS CALLED TO TRANSFER THE PROCESS TASK IMAGE INTO CORE. A PCB
                                        ; WILL BE CREATED IF THE PARTITION IN USE IS A SYSTEM CONTROLLED PARTITION.
                                        ; THE PRESENCE OF AN MCB PROCESS IN A SYSTEM CONTROLLED PARTITION IS DENOTED
                                        ; BY SETTING THE PARTITION NAME TO 'NT.***', WHERE '***' IS THE PDV NAME.
                                        ;
                                        ; INPUTS:
                                        ;       R1=FILE SPEC ADDRESS
                                        ;       R2=FILE SPEC LENGTH
                                        ;       $$PAR=PARTITION NAME (OPTIONAL)
                                        ;       $$INC=EXTENSION VALUE IN 32. WORD BLOCKS (OPTIONAL)
                                        ;       $NAME=RAD50 PDV NAME
                                        ;
                                        ; OUTPUTS:
                                        ;       C-BIT=SUCCESS/FAILURE
                                        ;       R0=ERROR CODE
                                        ;       R1=PARTITION CONTROL BLOCK ADDRESS
                                        ;       R2=DESTROYED
                                        ;-
 000000                                 $PRIO::
 000000 005067  000252'                         CLR     PCBA                    ; NO PCB YET
 000004 005067  000250'                         CLR     ERROR                   ; NO ERROR, EITHER
 000010                                         CALL    $CLOPE                  ; OPEN PROCESS FILE
 000014 105777  000000G                         TSTB    @$CLIOS                 ; LOOK AT GCL IOSB
 000020 100466                                  BMI     101$                    ; IF MI, FAILURE
 000022 052010                                  BIS     (R0)+,(R0)              ; IS FILE CONTIGUOUS?
 000024 001470                                  BEQ     111$                    ; IF EQ, NO
 000026                                         CALL    PR.LBL                  ; READ LABEL BLOCK
 000032 103405                                  BCS     10$                     ; IF CS, ERROR
 000034                                         CALL    PR.PAR                  ; FIND PROCESS PARTITION
 000040 103402                                  BCS     10$                     ; IF CS, ERROR
 000042                                         CALL    PR.XFR                  ; READ PROCESS TASK IMAGE INTO CORE
 000046                                 10$:    CALL    $CLDEQ                  ; CLOSE PROCESS FILE
 000052 005767  000250'                         TST     ERROR                   ; ANY ERRORS? (CLEARS C-BIT)
 000056 001411                                  BEQ     30$                     ; IF ZERO, NO
 000060 016700  0C0252'                         MOV     PCBA,R0                 ; GET PCB ADDRESS
 000064 001405                                  BEQ     20$                     ; IF ZERO, NOTHING TO CLEANUP
 000066                                         CALL    $XPAR                   ; RELEASE PARTITION
 000072 103451                                  BCS     121$                    ; IF CS, VERY BAD
 000074 005067  000252'                         CLR     PCBA                    ; SAFETY PLAY
 000100 000261                          20$:    SEC                             ; INDICATE FAILURE
 000102 016700  000250'                 30$:    MOV     ERROR,R0                ; GET ERROR CODE
 000106 016701  000252'                         MOV     PCBA,R1                 ; AND PCB ADDRESS
 000112 103430                                  BCS     40$                     ; DONE IF C-BIT SET
 000114 032767  000020  000000G                 BIT     #FE.PLA,$FMASK          ; IS THIS A PLAS SYSTEM?
 000122 001416                                  BEQ     35$                     ; NO IF EQ
 000124                                         GETRV   R1,,PLGTH               ; RETRIEVE PCB
 000142 016702  000000G                         MOV     .PPRO,R2                ; PROTECTION WORD OFFSET IN PCB
 000146 012762  177777  000000G                 MOV     #-1,$BFR(R2)            ; DO NOT ALLOW PLAS ACCESS
 000154                                         PUTRC                           ; WRITE BACK
 000160                                 35$:
 000160 016746  000260'                         MOV     $LDSIZ,-(SP)            ; SAVE LOAD SIZE
 000164                                         CALL    CR.MC                   ; CHECK FOR MICROCODE PARTITION
 000170 012667  000260'                         MOV     (SP)+,$LDSIZ            ; RSTORE LOAD SIZE
```

```
002514  016703  000010G                  MOV     P.SUB+$BFR,R3     ; GET NEXT SUB-PCB
002520  001445                           BEQ     70$               ; IF ZERO, END OF LIST
002522                                   GETRV   R3                ; READ IN PCB
002532  016704  000014G                  MOV     P.REL+$BFR,R4     ; GET BIAS OF SUB-PARTITION
002536  160204                           SUB     R2,R4             ; GIVING SIZE OF HOLE
002540  166704  000142'                  SUB     BLOX,R4           ; IS THE HOLE BIG ENOUGH?
002544  103423                           BCS     60$               ; IF MI, NO                              ;[MP01]
002546  010116                           MOV     R1,(SP)           ; SAVE PREVIOUS SUB-PCB ADDRESS
002550                                   GETRV   R0                ; READ IN PCB
002560  010267  000014G                  MOV     R2,P.REL+$BFR     ; SET SUB PARTITION BIAS
002564  010367  000010G                  MOV     R3,P.SUB+$BFR     ; AND POINTER TO NEXT SUB-PCB IN LIST
002570                                   PUTRC                     ; REWRITE PCB
002574  032767  000000G 000000G          BIT     #FL.TOP,$FLAGS    ; TOP-DOWN ALLOCATION?
002602  001462                           BEQ     90$               ; IF EQ, NO - ALL DONE
002604  060467  000014G                  ADD     R4,P.REL+$BFR     ; ALLOCATE FROM TOP OF HOLE
002610                                   PUTRC                     ; REWRITE PCB
002614  010301                   60$:    MOV     R3,R1             ; COPY SUB-PCB ADDRESS
002616                                   GETRV   R1                ; READ IN PCB
002626  016702  000016G                  MOV     P.BLKS+$BFR,R2    ; GET SUB-PCB SIZE
002632  000726                           BR      50$               ;
002634                           70$:    GETRV   R5                ; READ IN MAIN PCB
002644  016704  000014G                  MOV     P.REL+$BFR,R4     ; GET MAIN PARTITION BIAS
002650  066704  000016G                  ADD     P.BLKS+$BFR,R4    ; GIVING BIAS OF END OF PARTITION
002654  032767  000000G 000000G          BIT     #VF.SS,$VFLAG     ; 11S SYSTEM ?
002662  001006                           BNE     75$               ; IF YES - BRANCH

                                         .IF NDF R$$MPL                                                     ;[MP01]

002664  026727  000000G 000016           CMP     $ICBLN,#16        ; 11M SYSTEM ?
002672  001002                           BNE     75$               ; IF NO - BRANCH
002674  012704  007600                   MOV     #7600,R4          ; ALLOW ACCESS OF 124KW FOR 11M SYSTEMS
                                         .ENDC                                                              ;[MP01]

002700  160204                   75$:    SUB     R2,R4             ; GIVING SIZE OF LAST HOLE
002702  166704  000142'                  SUB     BLOX,R4           ; IS LAST HOLE BIG ENOUGH?
002706  103420                           BCS     90$               ; IF MI, NO                              ;[MP01]
002710  032767  000000G 000000G          BIT     #FL.TOP,$FLAGS    ; TOP-DOWN ALLOCATION?
002716  001401                           BEQ     80$               ; IF EQ, NO
002720  060402                           ADD     R4,R2             ; ALLOCATE FROM TOP OF HOLE
002722  010116                   80$:    MOV     R1,(SP)           ; SAVE PREVIOUS SUB-PCB ADDRESS
002724                                   GETRV   R0                ; READ IN PCB
002734  010267  000014G                  MOV     R2,P.REL+$BFR     ; SET SUB PARTITION BIAS
002740  010367  000010G                  MOV     R3,P.SUB+$BFR     ; AND POINTER TO NEXT SUB-PCB IN LIST
002744                                   PUTRC                     ; REWRITE PCB
002750  012601                   90$:    MOV     (SP)+,R1          ; GET PREVIOUS SUB-PCB ADDRESS
002752  001416                           BEQ     110$              ; IF EQ, NO SPACE IN PARTITION
002754                                   GETRV   R1                ; READ IN PCB
002764  010067  000010G                  MOV     R0,P.SUB+$BFR     ; LINK OUR SUB-PCB TO PREVIOUS
002770                                   PUTRC                     ; REWRITE PCB
002774  010005                           MOV     R0,R5             ; THIS IS NOW THE OFFICIAL PCB
002776                                   RETURN                    ;

                                 ;
                                 ; ERROR CONDITIONS
                                 ;
003000  012767  000036  000250'  100$:   MOV     #ERR30,ERROR      ; SUB-PCB ALLOCATION FAILURE
003006                                   RETURN                    ;
```

```
                        .IF NDF $GOFF
                        .TITLE  PRMPTS - VNP PROMPT STRINGS
                        .IFF
                        .TITLE  PRMPTO - VNP PROMPT STRING OFFSETS ONLY
                        .ENDC

                        .IDENT  /V05.00/                                           ;**-1

                ;
                ; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                ;
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                ; TRANSFERRED.
                ;
                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                ; CORPORATION.
                ;
                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                ;
                ;
                ;
                ; MODULE DESCRIPTION:
                ;
                ;       VNP - COMMAND PROMPT STRINGS
                ;
                ;
                ;
                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                ;
                ; IDENT HISTORY:
                ;
                ; 1.00  14-DEC-79
                ;       DECNET-11M/S V3.0
                ;       DECNET-11M-PLUS V1.0
                ;
                ; 3.00  16-APR-82
                ;       DECNET-11M V3.1
                ;       DECNET-11M-PLUS V1.1
                ;
                ; 4.00  07-NOV-83
                ;       DECNET-11M V4.0
                ;       DECNET-11M-PLUS V2.0
                ;
                ; 5.00  22-JUL-85
                ;       DECnet-11M/S V4.2
                ;       DECnet-11M-Plus V3.0
                ;       DECnet-Micro/RSX V1.0
                ;
                ;
```

```
000042                          PVAL:   .BLKW   1


                                ; START OF PROCESS VECTOR TABLE

000044  000056' 000000          VECSYM: .WORD   VLNSYM,0
000050                          VTSYM:  .BLKW   2
000054  000000                  VECVAL: .WORD   0

                                ; VECTOR TABLE SIZE (WORDS)

000056  000070' 000000          VLNSYM: .WORD   $PRINC,0
000062                          VLSYM:  .BLKW   2
000066  000000                  VLNVAL: .WORD   0


                                ;
                                ; PROCESS LINE TABLE SIZE
                                ;
000070  000102' 000000          $PRINC::.WORD   $PREXT,0
000074  126404  046166                  .RAD50  "$$LLIN"
000100                          $PRLLN::.BLKW   1

                                ;
                                ; PROCESS CONTROLLER TABLE SIZE
                                ;
000102  000114' 000000          $PREXT::.WORD   DTBLK,0
000106  126404  012404                  .RAD50  "$$LCNT"
000112                          $PRCLN::.BLKW   1

                                ;
                                ; DCB DISPATCH TABLE ADDRESS
                                ;
000114  000126' 000000          DTBLK:  .WORD   SQBLK,0
000120  126230  076534          DTSYM:  .RAD50  "$XXTBL"
000124                          DTVAL:  .BLKW   1

                                ;
                                ; FORK AND TIMER BLOCKS INITIALIZATION INFORMATION ( FOR AUX)
                                ;
000126  000140' 000000          SQBLK:  .WORD   MILBLK,0
000132  125711  074646                  .RAD50  "$SQSRV"
000136                          $SQSRV: .BLKW   1

000140  000152' 000000          MILBLK: .WORD   SECBLK,0
000144  063357  140075                  .RAD50  "PR100M"
000150                          PR100M: .BLKW   1

000152  000164' 000000          SECBLK: .WORD   XBLK1,0
000156  063357  073470                  .RAD50  "PR1SC"
000162                          PR1SC:  .BLKW   1

                                ;
                                ; COMM EXEC SYMBOLS USED FOR PROCESS VALIDATION
                                ;
000164  000176' 000000          XBLK1:  .WORD   XBLK2,0
000170  124473  007544                  .RAD50  "$CCBRT"
```

```
002212  160105                    SUB     R1,R5               ; ADJUST REMAINING SIZE IN FIRST BLK
002214                            PUTAD   R0,R1,R2            ; WRITE OUT CACHE BLOCK
002234  060100                    ADD     R1,R0               ; ADJUST ADDRESS
002236  010501                    MOV     R5,R1               ; COPY TRANSFER SIZE
002240  001401                    BEQ     60$                 ; IF NO MORE - BRANCH
002242  000750                    BR      30$                 ; CONTINUE

002244  010301            60$:    MOV     R3,R1               ; COPY SIZE OD SECOND 4KW BLOCK
002246  010305                    MOV     R3,R5               ; ...
002250  001467                    BEQ     120$                ; IF NO MORE - BRANCH
002252  005003                    CLR     R3                  ; FLAG SECOND BLOCK TRANSFER INITIATED
002254  062702  000200            ADD     #200,R2             ; ADJUST BIAS TO NEXT 4KW
002260  012700  140000            MOV     #140000,R0          ; SET VIRTUAL ADDRESS TO FIRST WORD
002264  000737                    BR      30$

002266  010105            70$:    MOV     R1,R5               ; SAVE SIZE
002270  022705  001000    80$:    CMP     #512.,R5            ; MORE THAN ONE TRANSFER
002274  002002                    BGE     90$                 ; IF NO - BRANCH
002276  012701  001000            MOV     #512.,R1            ; IF YES - LIMIT TRANSFER SIZE
002302  012703  000000G   90$:    MOV     #$BFR,R3            ; GET BUFFER ADDRESS
002306  010104                    MOV     R1,R4               ; COPY SIZE

002310  105023            100$:   CLRB    (R3)+               ; INIT BUFFER
002312  005304                    DEC     R4                  ; ANY LEFT ?
002314  001375                    BNE     100$                ; IF YES - BRANCH
002316                            PUTAD   R0,R1,R2            ; WRITE CACHE TO IMAGE
002336  022705  001000            CMP     #512.,R5            ; ANY LEFT TO TRANSFER ?
002342  002032                    BGE     120$                ; IF NO - BRANCH
002344  162705  001000            SUB     #512.,R5            ; IF YES - ADJUST REMAINING SIZE
002350  062700  001000            ADD     #512.,R0            ; ADJUST ADDRESS
002354  010501                    MOV     R5,R1               ; SAVE SIZE
002356  000744                    BR      80$                 ; CONTINUE


002360  012700  114224    110$:   MOV     #^RXPT,R0           ; GET XPT PDV ADDRESS
002364                            CALL    FNDPDV              ; ...
002370  103420                    BCS     130$                ; IF NOT PDV - BRANCH ??

002372  016000  000000G           MOV     Z.DAT(R0),R0        ; GET XPT DDB ADDRESS
002376                            GETRV   R0,#N$XLEN          ; READ XPT DDB
002414  005067  000062G           CLR     $BFR+N$CACH         ; INIT CACHE SIZE FOR ROUTING NODE
002420  005067  000064G           CLR     $BFR+N$CACH+2       ; INIT CACHE ADDRESS FOR ROUTING NODE
002424                            PUTRC                       ; WRITE XPT DDB TO IMAGE

002430                    120$:   RETURN

002432                    130$:   EMSG$R  70                  ; XPT NOT FOUND ?

```

```
                                                               L 8
VPRO - VNP SET PROCESS ROUTINES MACRO V05.03b Wednesday 28-Aug-85 20:04  Page 19-3
Symbol table

SF.ACT= 000200        T$LTPR  000014        XPTDSP  000016R   002 $ERR51= ****** GX     $PUTRC= ****** GX
SF.ENA= 000100        T$LTPS  000020        XPTPDV  000014R   002 $ERR53= ****** GX     $RADDR= ****** GX
SF.LPB= 000004        T$NAPL  000004        X$$DBT= 000000        $ERR54= ****** GX     $RBLCK= ****** GX
SF.MFL= 000040        T$NFE   000000        Z.AVL = ****** GX     $ERR55= ****** GX     $ROL    001452RG
SF.PAC= 000020        T$NLEN  000010        Z.DAT = ****** GX     $ERR56= ****** GX     $SAVAL= ****** GX
SF.REA= 000010        T$NNUL  000002        Z.DSP = ****** GX     $ERR57= ****** GX     $SLTA = ****** GX
SF.SER= 000001        T$NOPL  000006        Z.LLN = ****** GX     $ERR58= ****** GX     $SLTMA= ****** GX
SF.SVC= 000002        T$NRNI  000042        Z.MAP = ****** GX     $ERR59= ****** GX     $SQSRV  000136R   002
SF.UNL= 000040        T$NRPL  000005        Z.NAM = ****** GX     $ERR60= ****** GX     $SYMBL= ****** GX
SQBLK   000126R   002 T$NRUL  000007        Z.PCB = ****** GX     $ERR61= ****** GX     $T1SCL= ****** GX
STBERR  001054R       T$NVR   000001        $ALADJ  002440RG      $ERR62= ****** GX     $T100C= ****** GX
STBEXT  000020R   002 T$RPRI  000040        $ALECL  004056RG      $ERR63= ****** GX     $XFER   000000RG
SYSFDB= ****** GX     T$SVC   000034        $ALMHC  003424RG      $ERR64= ****** GX     $$DEV = ****** GX
S$$WRG= 000000        T$T5    000030        $ALN18= ****** GX     $ERR65= ****** GX     $$HTIM= ****** GX
S$$YSZ= 007600        T$T6    000032        $ALPOL= ****** GX     $ERR66= ****** GX     $$LTIM= ****** GX
S.COST  000001        T$$KMG= 000000        $ALTLC  001062RG      $ERR67= ****** GX     $$NBRA= ****** GX
S.FLG   000000        T$$MIN= 000000        $AZFS = ****** GX     $ERR69= ****** GX     $$NODC= ****** GX
S.LEN   000004        VECLEN  000006R   002 $BFR  = ****** GX     $ERR70= ****** GX     $$RPRI= ****** GX
S.NMST  000002        VECSYM  000044R   002 $BIAS = ****** GX     $ERR9D= ****** GX     $$TBL = ****** GX
S.OWNR  000003        VECTAB  000000R   002 $CACH   002000RG      $ERR99= ****** GX     .CXALL  000062
TSKERR  000340R       VECVAL  000054R   002 $CAT5 = ****** GX     $GETAD= ****** GX     .CXCSR  000004
T$FLAG  000044        VF.LIB= 000001        $CEACC= ****** GX     $GETRV= ****** GX     .CXLBR  000060
T$LIF   000013        VLNSYM  000056R   002 $CETDF= ****** GX     $LDSIZ= ****** GX     .CXLB1  000030
T$LIFL  000013        VLNVAL  000066R   002 $CKT    002720RG      $LLCTA= ****** GX     .CXLB2  000044
T$LIFO  000013        VLSYM   000062R   002 $CMFRK= ****** GX     $MUL  = ****** GX     .CXLEN  000064
T$LIFS  000013        VTSYM   000050R   002 $CTSYM  004504RG      $NAME = ****** GX     .CXPCB  000006
T$LIN   000000        V$$CTR= 001000        $DECPT= ****** GX     $NTLHB= ****** GX     .CXSYM  000010
T$LIPS  000006        V.ADD   000006        $DFN  = ****** GX     $PDVA = ****** GX     .CXUNL  000024
T$LLD   000012        V.BIAS  000004        $DFNAM= ****** GX     $PDVNM= ****** GX     .CXVEC  000026
T$LLDC  000045        V.FLG   000002        $ERRQX= ****** GX     $PDVTA= ****** GX     .CXVSM  000016
T$LLDL  000012        V.LEN   000012        $ERR44= ****** GX     $FRCLN  000112RG  002 .MGMGE= ****** GX
T$LLDO  000012        V.LNK   000000        $ERR45= ****** GX     $FREXT  000102RG  002 .NTGCD  000000
T$LLDS  000012        V.PDV   000004        $ERR46= ****** GX     $PRINC  000070RG  002 .NTPCB  000000
T$LLEN  000046        V.SIZ   000010        $ERR47= ****** GX     $PRIO = ****** GX     .NTUMR  000002
T$LOPR  000002        XBLK1   000164R   002 $ERR49= ****** GX     $PRLLN  000100RG  002 .PLGTH= ****** GX
T$LTCL  000024        XBLK2   000176R   002 $ERR50= ****** GX     $PUTAD= ****** GX     .SBPCB  000066
T$LTIM  000026        XBLK3   000210R   002

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        004504    001   (RW,I,LCL,REL,CON)
DATA    000274    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 52
Work  file writes: 54
Size of work file: 18984 Words  ( 75 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:55.23
DB2:VPRO.T34,[132,134]VPRO/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VPRO
```

M 8

```
     58                                 ;       DECnet-Micro/RSX V1.0
     59                                 ;
     60                                 ;
     61                                 ;
```

```
                        .SBTTL  DATA FOR INPUT PARAMETERS
                ;
                ; GENERAL INPUT PARAMETERS
                ;
002002          $FUNC:: .BLKW   1               ; FUNCTION CODE
002004          $OPTON::.BLKW   1               ; OPTIONS PRESENT
002006          $CMAND::.BLKW   1               ; COMMAND TYPE
002010          $QUAL:: .BLKW   1               ; COMMAND QUALIFIERS
002012          $TYPE:: .BLKW   1               ; SHOW COMMAND TYPE
002014          $MISS:: .BLKW   1               ; MISSING PARAMETERS BIT MASK


                        .IF NDF R$$MPL                                          ;[MP01]
        000004  $KRBLN  == 4                    ; SIZE OF NETWORK KRB

                        .IFTF                                                   ;[MP01]
        000016  $ICBI   == 16                   ; SIZE OF ICB                           ;[MP01]
002016  000016  $ICBLN::.WORD   $ICBI           ; DEFAULT TO I-ONLY ICB SIZE            ;[MP01]

                        .IFF                                                    ;[MP01]
                $KRBLN  == 26                   ; SIZE OF KRB                           ;[MP01]
                $ICBD   == 30                   ; SIZE OF I/D SPACE ICB                 ;[MP01]
                        .ENDC                                                   ;[MP01]

                ;
                ; $CMAND - BIT DEFINITIONS
                ;
        000001  CM$SET==1                       ; SET COMMAND
        000002  CM$CLE==2                       ; CLEAR COMMAND
        000004  CM$SHO==4                       ; SHOW COMMAND
                ;
                ; $MISS - BIT DEFINITIONS
                ;
        000001  MS.CSR==1                       ; CSR MISSING
        000002  MS.PRI==2                       ; INTERRUPT PRIORITY MISSING
        000004  MS.VCT==4                       ; VECTOR MISSING
        000010  MS.MTP==10                      ; MULTIPOINT STATION NUMBERS MISSING
        000420  MS.MUX==420                     ; MUX UNIT INFORMATION MISSING
        000437  MS.ALL==437                     ; ALL LINE INFO
        000040  MS.LLC==40                      ; LLC INFO MISSING
        000100  MS.INC==100                     ; LINE TABLE COUNT MISSING
        000200  MS.EXT==200                     ; CONTROLLER TABLE COUNT MISSING
        000400  MS.SEC==400                     ; SECONDARY CSR MISSING
        001000  MS.CIR==1000                    ; CIRCUIT PARAMETERS MISSING
                ;
                ; PROCESS PARAMETERS
                ;
002020          $FEATR::.BLKW   1               ; CURRENT LLC DEVICE FEATURES WORD
002022          $$EXT:: .BLKW   1               ; LLC - NUMBER OF 32. WORD BLOCKS TO EXTEND PROCESS
                                                ; DDM/DLC - NUMBER OF CONTROLLER TABLES IN PROCESS SPACE

                .IF DF  R$$MPL                                                  ;[MP01]
                .DBIAS::.BLKW   1               ; START OF D-DPACE                      ;[MP01]
                .ENDC                                                           ;[MP01]
002024          .NN::   .BLKW   1               ; NUMBER OF NODES MAXIMUM
002026          .MAXC:: .BLKW   1               ; MAXIMUM COST ALLOWED
002030          .MAXH:: .BLKW   1               ; MAXIMUM HOPS ALLOWED
```

```
   1095 004514  066703  173736                     ADD     $RADDR,R3          ; CALC OFFSET OF REFERENCED LOCN
   1096 004520  005202                     23$:    INC     R2                 ; FORM NEXT HIGHEST BLOCK NUMBER
   1097 004522  162703  001000                     SUB     #1000,R3           ; IS ADDRESS IN PREVIOUS BLOCK ?
   1098 004526  100374                             BPL     23$                ; IF PL NO
   1099 004530  005302                             DEC     R2                 ; FORM CORRECT BLOCK NUMBER
   1100 004532  062703  001000                     ADD     #1000,R3           ; FORM OFFSET IN BLOCK
   1101 004536  000410                             BR      27$                ; BRANCH TO VALIDATE AND PERFORM THE I/O
   1102 004540                             25$:
   1103                                            .IF DF  R$$MPL                                                   ;[MP01]
   1104                                            BIC     #DF.SCO,$DFLAG     ; DISABLE LOWER FLAG
   1105                                            .ENDC                                                            ;[MP01]
   1106
   1107                                                                       ; LABEL FOR NON-DISPLACED MEMORY XFER
   1108 004540  010203                             MOV     R2,R3              ; COPY ADDRESS OF RECORD
   1109 004542  042703  177000                     BIC     #^C<777>,R3        ; CLEAR ALL BUT OFFSET IN BLOCK
   1110 004546  040302                             BIC     R3,R2              ; CLEAR ALL BUT BLOCK NUMBER
   1111 004550  062702  003000                     ADD     #3*256.*2,R2       ; CALCULATE VIRTUAL BLOCK NUMBER
   1112 004554  000302                             SWAB    R2                 ; SWAP BLOCK NUMBER TO LOW BYTE
   1113 004556  006202                             ASR     R2                 ; CONVERT TO RELATIVE BLOCK NUMBER
   1114 004560                             27$:                               ; REF LABEL TO VALIDATE AND PERFORM I/O
   1115 004560  020260  000006                     CMP     R2,F.HIBK+2(R0)    ; LEGAL VIRTUAL BLOCK NUMBER ?
   1116 004564  101027                             BHI     30$                ; IF HI NO
   1117 004566                                     CALL    .POINT             ; POINT TO RECORD IN TASK IMAGE FILE
   1118 004572  103424                             BCS     30$                ; IF CS ERROR
   1119 004574  012603                             MOV     (SP)+,R3           ; RESTORE R3
   1120 004576  016060  000002  000024             MOV     F.RSIZ(R0),F.NRBD(R0) ; SET SIZE OF RECORD TO GET/PUT
   1121 004604  001407                             BEQ     28$                ; ILLEGAL RECORD SIZE
   1122 004606  026027  000002  001000             CMP     F.RSIZ(R0),#1000   ; MAXIMUM RECORD SIZE IS 512. BYTES
   1123 004614  101005                             BHI     29$                ; TOO BIG ...
   1124 004616                                     CALL    @(SP)+             ; ACCESS RECORD
   1125 004620  103421                             BCS     40$                ; IF CS I/O ERROR
   1126 004622                                     RETURN                     ;
   1127
   1128 004624  005026                     28$:    CLR     (SP)+              ; POP SUBROUTINE ADDRESS OFF STACK
   1129 004626                                     RETURN                     ; RETURN ( NULL TRANSFER SIZE )
   1130
   1131                                    ;
   1132                                    ;       ERROR PROCESSING
   1133                                    ;
   1134
   1135 004630                             29$:    ERRPT$  ERRV4,$EROUT,FATAL         ; ILLEGAL RECORD SIZE
   1136 004644  010267  000000'            30$:    MOV     R2,$BFR                    ; STORE ILLEGAL BLOCK NUMBER
   1137 004650                                     ERRPT$  ERRV1,$EROUT,FATAL
   1138 004664                             40$:    ERRPT$  ERRV2,$EROUT,FATAL         ; I/O ERROR
```

```
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 1

SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

ARAMSK  = 176000   G        #15-641
A.LULU  = 000002              6-102
CEACC     000500 RG         #6-127
CFSFLG    001026 RG         #7-211
CL.AST  = 004000            #6-102
CL.ATT  = 000200              6-102      #6-102
CL.CAS  = 020000            #6-102
CL.CMT  = 000001              6-102      #6-102
CL.EIF  = 000040            #6-102
CL.EXN  = 040000            #6-102
CL.EXT  = 000010              6-102      #6-102
CL.IND  = 000002              6-102      #6-102
CL.LCC  = 000020              6-102      #6-102
CL.NPR  = 000400            #6-102
CL.OPN  = 000100   G        #6-102        6-103
CL.RNE  = 000004            #6-102
CL.XIT  = 100000            #6-102
CM$CLE  = 000002   G        #15-479
CM$SET  = 000001   G        #15-478
CM$SHO  = 000004   G        #15-480
C.DSDS  = 000006              6-102       6-113
C.SIZE  = 000054              6-102
DEFFLG  = 000233            #6-102        6-102
DFCNT     001032 RG         #7-213
DFEXT     001606 R            8-249      #13-396
DFLTNB    000426 R            6-102      #6-102
DFPTR     001030 RG         #7-212
DFTLEN  = 000007            #6-102
DSIZE   = 000044   G        #7-152
D$END   = 000104              6-140
EF      = 000001            #6-102        6-102        6-102        6-102
ERRV1     005316 R            19-1137    #23-1285
ERRV2     005406 R            19-1138    #23-1287
ERRV3     005446 R            22-1272    #23-1289
ERRV4     005616 R            19-1135    #23-1292
FATAL   = 176031   G        #14-407       19-1135      19-1137      19-1138
FE.EXV  = 000004              18-1027
FILE      001374 RG           8-233       8-258       #13-389
FL.CEX  = 000010   G        #15-573
FL.CHA  = 000020   G        #15-574
FL.CNT  = 004000   G        #15-581
FL.DDM  = 000001   G        #15-570
FL.DLC  = 000002   G        #15-571
FL.ECH  = 000040   G        #15-575
FL.FDX  = 000100   G        #15-576
FL.HDX  = 000200   G        #15-577
FL.KMX  = 000400   G        #15-578
FL.LLC  = 000004   G        #15-572
FL.LMC  = 001000   G        #15-579
FL.MUX  = 002000   G        #15-580
FL.PX1  = 010000   G        #15-582
FL.PX2  = 020000   G        #15-583
```

```
VRES       CREATED BY  MACRO  ON 17-SEP-85 AT 16:16     PAGE 14
SYMBOL CROSS REFERENCE                                  CREF   04.00
SYMBOL  VALUE              REFERENCES

.ABTIM    003640 RG        #16-722
.ALLOC    001012 RG        #7-194
.CEAVL    003662 RG        #16-735
.CEPWR    003670 RG        #16-740
.CLINS    003572 RG        #16-702
.CLKHD    003644 RG        #16-724
.CRAVL    003552 RG        #16-694
.CRCOV    003666 RG        #16-739
.DEUMR    003634 RG        #16-720
.DEVHD    003556 RG        #16-696
.DLCOV    003664 RG        #16-738
.EXSIZ    003560 RG        #16-697
.FMASK    003562 RG        #16-698
.FRKHD    003574 RG        #16-703
.FSRCB  = ******  G          6-89
.GET    = ******  GX        18-1035     21-1200
.HEADR    003576 RG        #16-704
.MAXC     002026 RG        #15-506
.MAXH     002030 RG        #15-507
.MGEXT    003626 RG        #16-717
.MGMGE    003630 RG        #16-718
.NETPF    003654 RG        #16-728
.NN       002024 RG        #15-505
.NS0      003606 RG        #16-709
.NS1      003610 RG        #16-710
.NS2      003612 RG        #16-711
.NS3      003614 RG        #16-712
.NS4      003616 RG        #16-713
.NS5      003620 RG        #16-714
.NS6      003622 RG        #16-715
.NS7      003624 RG        #16-716
.NTUIC    003656 RG        #16-729
.PARHD    003564 RG        #16-699
.PATT     003646 RG        #16-725
.PDVTA    003660 RG        #16-734
.PHDR     003650 RG        #16-726
.PLGTH    003602 RG        #16-706
.POINT  = ******  GX        19-1117
.PPRO     003652 RG        #16-727
.PUT    = ******  GX        19-1080     20-1164
.RDBNM    001014 RG        #7-195
.RDBSZ    001016 RG        #7-196
.ROUTM    002032 RG        #15-508
.SDBNM    001020 RG        #7-197
.SDBSZ    001022 RG        #7-198
.STKDP    003600 RG        #16-705
.SYSIZ    003566 RG        #16-700
.SYSNM    003570 RG        #16-701
.TLGTH    003642 RG        #16-723
.TSKHD    003636 RG        #16-721
.UMRHD    003632 RG        #16-719
.VCTR     003604 RG        #16-708
```

```
                                                .SBTTL  RNTI,SCEX - CHECK SET SYS
                                        ;+
                                        ; SCEX - CHECK "SET CEX" REQUEST BLOCK
                                        ;
                                        ; INPUTS:
                                                R3=REQUEST BLOCK POINTER
                                                        .BLKW   1       ; LR.BLK - NETLDR DATA BLOCK ADDRESS
                                        ;
                                        ; OUTPUTS:
                                                C-BIT=SJCCESS/FAILURE
                                                DATA BLOCK ADDRESS STORED IN $NTLPT
                                                FL.CEX SET IN $FLAGS
                                        ;-
000124                                  SCEX:
000124  112767  000000G 000000G                 MOVB    #FL.CEX,$FLAGS  ; SET NETLDR MODE
000132  032767  000000G 000000G                 BIT     #FE.CEX,$FMASK  ; IS COMM EXEC ALREADY LOADED?
000140  001006                                  BNE     10$             ; IF NE, YES
000142  011367  000000G                         MOV     (R3),$$MUX      ; SAVE DATA BLOCK ADDRESS
000146  001004                                  BNE     20$             ; IF NE, NO PROBLEM
000150                                          EMSG$R  91              ; NO DATA BLOCK ADDRESS
000156  005004                            $:    CLR     R4              ; NO WORK TO DO!
000160                                  20$:    RETURN



                                        ;+
                                        ; RNTI  - BE SURE CEX IS UNLOADED
                                        ;
                                        ; INPUTS:
                                                R4 = FUNCTION CODE
                                        ;
                                        ;-

000162  032767  000000G 000000G RNTI:           BIT     #FE.CEX,$FMASK          ; IS COMEXEC ALREADY LOADED ?
000170  001401                                  BEQ     10$                     ; IF EQ, NO
000172  005004                                  CLR     R4                      ; ELSE, NO WORK TO DO !
000174                                  10$:    RETURN
```

```
    716                                         .SBTTL  CLRPP - CLEAR PROCESS PARMS
    717                                 ;+
    718                                 ; CLRPP - CLEAR PROCESS PARAMETERS
    719                                 ;
    720                                 ; INPUTS:
    721                                 ;       NONE
    722                                 ;
    723                                 ; OUTPUTS:
    724                                 ;       $$INC=0
    725                                 ;       $$PAR=0
    726                                 ;-
    727 001532                          CLRPP:
    728 001532  005067  000000G                 CLR     $$INC                   ; SET NO PROCESS INCREMENT
    729 001536  005067  000000G                 CLR     $$PAR                   ; AND NO PARTITION OVERRIDE
    730 001542                                  RETURN
```

```
                                        .SBTTL  $RERR - PRINT ERROR MESSAGE
                                ;+
                                ; $RERR - PRINT ERROR MESSAGE
                                ;
                                ; INPUTS:
                                ;       R0=MESSAGE BLOCK ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       C-BIT=SET
                                ;       R0,R1=DESTROYED
                                ;-
003250                          $RERR::
003250                                  CALL    $SAVAL          ; SAVE ALL REGISTERS
003254  010005                          MOV     R0,R5           ; COPY ERROR MESSAGE BLOCK ADDRESS
003256  012704  000400'                 MOV     #ERRBUF,R4      ; PLACE TO BUILD ERROR MESSAGE
003262  012503                          MOV     (R5)+,R3        ; GET FORMAT STRING ADDRESS
003264  005775  000000                  TST     @(R5)           ; JUST IN CASE...
003270  001423                          BEQ     40$             ; BRANCH IF NO FORMATTING
003272                                  CALL    @(R5)+          ; DO THE FORMATTING
003274  105713                          TSTB    (R3)            ; IS IT END OF FORMAT STRING?
003276  001403                          BEQ     20$             ; IF EQ, YES
003300  112324                  10$:    MOVB    (R3)+,(R4)+     ; ELSE COPY REST OF FORMAT STRING
003302  001376                          BNE     10$             ; ...
003304  005304                          DEC     R4              ; SKIP THE ZERO BYTE
003306  012503                  20$:    MOV     (R5)+,R3        ; GET LINE 2 ADDRESS
003310  112524                  30$:    MOVB    (R5)+,(R4)+     ; COPY THE ERROR TEXT
003312  001376                          BNE     30$             ; ...
003314  005304                          DEC     R4              ; SKIP THE ZERO BYTE
003316  012700  000370'                 MOV     #ERRMSG,R0      ; GET ERROR MESSAGE ADDRESS
003322  010401                          MOV     R4,R1           ; COPY END ADDRESS
003324  160001                          SUB     R0,R1           ; GIVING LENGTH
003326                                  CALL    $TIOUT          ; PRINT IT
003332  005703                          TST     R3              ; SHOULD SECOND LINE BE PRINTED?
003334  001401                          BEQ     40$             ; IF EQ, NO
003336                                  CALL    (R3)            ; PRINT SECOND ERROR MESSAGE LINE
003340  000261                  40$:    SEC                     ; RETURN WITH C-BIT SET
003342                                  RETURN
```

```
                                                                  M  1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50      PAGE 9
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE               REFERENCES

LX.CEX  = 000004            #5-78         16-963
LX.CIR  = 000034            #5-78
LX.LIN  = 000006            #5-78
LX.LOG  = 000021            #5-78
LX.NOD  = 000015            #5-78
LX.OBJ  = 000023            #5-78
LX.PRO  = 000005            #5-78
L.COST    000015            #5-73
L.CTL     000012            #5-73         59-2688     97-4227     107-4785    108-4823
L.CVA     177776            #5-73
L.DDM     000002            #5-73         59-2692     97-4225     106-4748
L.DDS     000004            #5-73
L.DLC     000003            #5-73
L.DLM     000006            #5-73
L.DLS     000010            #5-73
L.FLG     000000            #5-73        *28-1695    *59-2701     59-2703     59-2705     59-2712    *59-2714     66-3002     106-4746
L.KRBA    000016            #5-73
L.LEN   = 000022            #5-73
L.LGST    013206 R          #81-3559
L.LNG     000124             24-1429
L.MPF     000022            #5-73         59-2707     66-2999     108-4812
L.NMST    000020            #5-73
L.NSTA    000014            #5-73         66-2993     108-4809
L.OWNR    000021            #5-73
L.UNT     000013            #5-73         59-2690     97-4229     107-4786    108-4824
MACTYP  = ******   GX       *31-1874
MAXCST  = 001777            #7-200        71-3261
MAXHOP  = 000037            #7-201        72-3286
MC.ADD  = 000200            #27-1656
MC.DDM  = 000004            #27-1651
MC.DLC  = 000003            #27-1650
MC.EVT  = 001000            #27-1658
MC.LLC  = 000002            #27-1649
MC.LOG  = 000400            #27-1657
MC.NOD  = 000010            #27-1652
MC.OBJ  = 000020            #27-1653
MC.REM  = 000040            #27-1654
MC.ROU  = 000100            #27-1655
MC.SLT  = 000001            #27-1648
MC.STA  = 002000            #27-1659
MOVBUF    016016 R          #100-4361     101-4400    101-4404    101-4409
MPAR    = 172100            #5-75
MPCSR   = 177746            #5-75
MRKFL     000040 R          #6-122       *28-1704     59-2701     66-3001
MXCP      000064 R          #6-131       *82-3583    *86-3668     86-3683
M$$MGE  = 000001            #5-68         5-75
M.BFVH    000011            *11-718
M.BFVL    000012            *11-717
M.LGTH  = 000014             6-117        11-692      11-742      16-1019
M.UMRA    000002             11-732
M.UMRN    000004            *11-708
NA        000076 R          #6-139       *24-1331     24-1399
```

```
MACRO NAME     REFERENCES

ADJDF$         #5-58      5-62
ASL$           #5-56
ASR$           #5-55
CALL            8-221      8-236      8-239      8-270      8-287      8-293      8-298      8-317      8-329      8-337
                8-362      8-368      8-402      8-418      8-448      8-452      8-457      8-459      8-462      8-467
                8-472      8-483      8-492      8-499      8-503      8-511      9-541      10-583     10-600     11-693
                11-730     11-742     12-786     12-797     13-834     13-839     14-879     14-884     16-948     16-950
                16-964     16-972     16-976     16-1018    16-1020    17-1054    17-1103    24-1355    24-1364    24-1388
                24-1417    24-1430    24-1443    24-1478    24-1483    26-1535    26-1569    26-1573    26-1577    26-1583
                26-1587    26-1592    26-1594    26-1597    26-1607    26-1611    26-1614    26-1616    26-1630    28-1682
                28-1683    28-1700    28-1702    28-1705    28-1707    29-1729    29-1730    30-1752    30-1769    30-1772
                30-1775    30-1784    30-1790    30-1799    30-1805    30-1812    30-1819    30-1830    30-1846    31-1875
                55-2624    61-2753    61-2778    62-2792    62-2801    62-2808    62-2821    63-2841    63-2844    63-2863
                63-2868    64-2887    64-2889    64-2894    65-2920    65-2925    65-2926    65-2955    65-2956    65-2960
                67-3017    67-3020    67-3027    67-3030    67-3037    67-3039    67-3046    67-3048    67-3056    67-3058
                67-3065    67-3067    67-3074    67-3076    67-3083    67-3085    67-3093    67-3095    67-3103    67-3105
                67-3110    67-3112    69-3163    69-3174    69-3175    69-3178    69-3181    69-3184    69-3188    69-3191
                70-3239    70-3243    71-3257    71-3266    72-3280    72-3291    73-3303    73-3306    74-3318    74-3330
                75-3344    75-3357    75-3372    75-3376    76-3392    76-3396    76-3408    76-3420    79-3494    79-3505
                79-3506    79-3512    79-3514    80-3541    80-3542    83-3606    83-3610    86-3675    86-3685    86-3686
                86-3693    86-3695    87-3725    87-3726    88-3747    88-3761    88-3762    91-3907    92-3937    92-3938
                92-3946    93-3991    93-3999    93-4010    94-4026    94-4033    94-4060    94-4063    94-4073    94-4076
                95-4115    95-4117    98-4258    98-4271    98-4272    98-4281    101-4390   101-4400   101-4404   101-4409
                102-4433   102-4450   102-4451   102-4454   102-4464   102-4467   103-4512   103-4513   103-4516   103-4523
                103-4524   103-4526   103-4529   103-4530   103-4536   103-4545   103-4549   103-4553   105-4706   105-4707
                105-4709   105-4713   106-4726   106-4738   106-4753   106-4756   107-4789   108-4827   109-4861   110-4877
CALLR          #5-58
CCBDF$         #5-55      5-71
CEACC$         #5-57      65-2925    65-2955    69-3174    79-3505    80-3541    86-3685    87-3725    88-3761    94-4033
                98-4271   102-4450   103-4523   103-4529   105-4706
CTRDF$         #5-58      5-69
DBGTP$         #18-1108   #19-1139   #20-1178   #20-1184   #20-1189   #20-1196   #20-1199   #20-1202   #20-1205   #20-1208
               #20-1213   #21-1220   #21-1233   #21-1234   #22-1250   #23-1265   #23-1283   #23-1286   #23-1287   #23-1288
               #23-1291   #23-1300   #23-1303   #23-1306   #23-1309   #23-1312   #23-1315   #23-1318   #32-1884   #32-1892
               #32-1893   #32-1894   #32-1895   #32-1896   #32-1897   #32-1898   #32-1899   #32-1900   #32-1901   #32-1902
               #32-1903   #32-1904   #32-1905   #32-1906   #32-1907   #32-1908   #33-1918   #33-1921   #34-1935   #34-1936
               #34-1937   #34-1938   #34-1943   #34-1948   #35-1966   #35-1967   #35-1968   #35-1969   #35-1970   #35-1979
               #35-1982   #35-1985   #35-1988   #35-1991   #35-1994   #35-1995   #35-1998   #36-2017   #36-2020   #36-2021
               #36-2022   #36-2023   #36-2024   #39-2078   #39-2100   #42-2226   #42-2227   #42-2237   #45-2296   #45-2314
               #46-2329   #46-2334   #46-2335   #46-2339   #46-2343   #46-2347   #46-2351   #46-2355   #46-2359   #46-2363
               #47-2375   #47-2376   #47-2383   #47-2386   #47-2387   #47-2388   #48-2406   #48-2407   #48-2414   #48-2418
               #48-2422   #48-2429   #48-2433   #48-2437   #48-2447   #48-2451   #48-2455   #48-2459   #49-2468   #49-2472
               #49-2476   #49-2480   #49-2484   #49-2488   #49-2492   #49-2497   #49-2510
DHBDF$         #5-58      5-66
DIR$           #5-70      9-545      9-554      9-560      10-587     10-595     10-606
ERBLK$         #8-219     #110-4883
ERMSG$         #8-219     #110-4883
ERRDEF         #5-98      10-629     10-630     10-631     10-632     10-633     10-634     10-635     10-636     10-637
                10-638     10-639     10-640     10-641     10-642     10-643     10-644     10-645     10-646     10-647
                10-648     10-649     10-650     10-651     10-652     10-653     10-654     10-655     10-656     10-657
                10-658     10-659     10-660     10-661     10-662     10-663     10-664     10-665
ERRPT$         #8-219     #110-4883
```

```
    300
    301 000174                                  40$:    RETURN
    302
    303                                         ;
    304                                         ; ERROR CONDITIONS
    305                                         ;
    306 000176  012767  000002  000250'         101$:   MOV     #ERR2,ERROR     ; OPEN FAILURE
    307 000204  000735                                  BR      20$
    308 000206  012767  000006  000250'         111$:   MOV     #ERR6,ERROR     ; NOT CONTIGUOUS
    309 000214  000714                                  BR      10$
    310 000216                                  121$:   CRASH                   ; SOMEBODY UNLINKED OUR SUB-PCB!
    868 003010  012767  000040  000250' 110$:   MOV     #ERR32,ERROR    ; MAIN PARTITION TOO FRAGMENTED
    869 003016  016701  000000G                 MOV     .PLGTH,R1       ; GET PCB LENGTH
    870 003022                                  CALLR   $DEA16          ; DE-ALLOCATE SUB-PCB
```

```
     58                                                                                              ;**-1
```

```
000174  000000                          .WORD   0
000176  000210' 000000          XBLK2:  .WORD   XBLK3,0
000202  124473  006654                  .RAD50  "$CCBGT"
000206  000000                          .WORD   0
000210  000000  000000          XBLK3:  .WORD   0,0
000214  126344  035115                  .RAD50  "$ZTIME"
000220  000000                          .WORD   0



                                ;
                                ; SYMBOL TABLE ERRORS
                                ;
000222  000000G                 ETAB1:  .WORD   $ERR44          ; OPEN FAILURE
000224  000000G                         .WORD   $ERR45          ; READ FAILURE
000226  000000G                         .WORD   $ERR46          ; READ OVERFLOW
000230  000000G                         .WORD   $ERR47          ; BAD STB FORMAT

                                ;
                                ; PROCESS I/O ERRORS
                                ;
000232  000000G                 ETAB2:  .WORD   $ERR51          ; OPEN FAILURE
000234  000101                          .WORD   101             ; GET ATTRIBUTES FAILED
000236  000000G                         .WORD   $ERR53          ; FILE NOT CONTIGUOUS
000240  000000G                         .WORD   $ERR54          ; LABEL BLOCK READ FAILURE
000242  000000G                         .WORD   $ERR55          ; INVALID IMAGE (HEADER, OVERLAID, ETC.)
000244  000000G                         .WORD   $ERR56          ; PROCESS NOT BUILD FOR APR5
000246  000000G                         .WORD   $ERR57          ; PROCESS EXCEEDS 4K WORDS
000250  000000G                         .WORD   $ERR58          ; PROCESS EXTENSION EXCEEDS 4K WORDS
000252  000000G                         .WORD   $ERR59          ; PARTITION NOT IN SYSTEM
000254  000000G                         .WORD   $ERR60          ; BASE ADDRESS MISMATCH
000256  000000G                         .WORD   $ERR61          ; PARTITION TOO SMALL
000260  000000G                         .WORD   $ERR62          ; NOT COMMON PARTITION
000262  000000G                         .WORD   $ERR63          ; PARTITION BUSY
000264  000000G                         .WORD   $ERR64          ; CEX PARTITION NOT IN EXEC SPACE
000266  000000G                         .WORD   $ERR65          ; SUB-PCB ALLOCATION FAILURE
000270  000000G                         .WORD   $ERR66          ; MAIN PARTITION TOO FRAGMENTED
000272  000000G                         .WORD   $ERR67          ; PROCESS READ FAILURE

000000                                  .PSECT
```

```

                                               .SBTTL  $ALADJ  - ALLOCATE ADJACENCY DATA BASE

                                      ;+
                                      ; $ALADJ - ALLOCATE ADJACENCY DATA BASE
                                      ;
                                      ; INPUTS:
                                      ;        $PDVA - XPT PDV ADDRESS
                                      ;
                                      ; OUTPUTS:
                                      ;        ADJACENCY DATA BASE ALLOCATED IF ROUTING NODE
                                      ;
                                      ;-

002440                                $ALADJ::CALL    $SAVAL            ; SAVE REG'S
002444  016703  000000G                       MOV     $PDVA,R3          ; GET PDV ADDRESS
002450  005000                                CLR     R0                ; CLEAR REG
002452  156300  000000G                       BISB    Z.LLN(R3),R0      ; GET XPT CHANNEL COUNT
002456  005200                                INC     R0                ; INCLUDE LOOPBACK CHANNEL
002460                                        GETRV   $DECPT,#D$END     ; READ DECNET HOME BLOCK
002500  005767  000060G                       TST     D$NN+$BFR         ; ENDNODE ?
002504  001501                                BEQ     10$               ; IF YES - BRANCH
002506  066700  000054G                       ADD     D$NBRA+$BFR,R0    ; INCLUDE BROADCAST ROUTERS
002512  066700  000056G                       ADD     D$NBEA+$BFR,R0    ; INCLUDE BROADCAST ENDNODE ADJACECIES
002516  016704  000014G                       MOV     D$LNUM+$BFR,R4    ; COPY NODE ADDRESS

002522  012701  000004                        MOV     #A$LEN,R1         ; STORE ADJACENCY ENTRY LENGTH
002526                                        CALL    $MUL              ; CALCULATE TOTAL SIZE
002532  062701  000077                        ADD     #77,R1            ; ROUND UP TO NEXT BLOCK
002536  042701  000077                        BIC     #77,R1            ; ...
002542  000241                                CLC                       ; CLEAR C-BIT
        000006                                .REPT   6
                                              ASR     R1                ; CONVERT TO BLOCKS
                                              .ENDM

002560                                        CALL    $ALN18            ; ALLOCATE POOL..
002564  103452                                BCS     101$              ; IF FAILURE - BRANCH
002566                                        GETRV   Z.DAT(R3),#N$XLEN ; READ XPT DDB
002606  010067  000072G                       MOV     R0,N$ADJ1+$BFR    ; STORE ADJACENCY START
002612                                        PUTRC
002616                                        GETAD   #120000,#2,R0     ; READ A WORD FROM ADJACENCY DATA BASE
002642  010467  000000G                       MOV     R4,$BFR           ; COPY NODE ADDRESS
002646                                        PUTAD                     ; WRITE THE WORD TO IMAGE FILE

002652                                        CALL    $ALN18            ; ALLOCATE SECOND SECTION OF ADJACENCY D.B.
002656  103415                                BCS     101$              ; IF FAILURE - BRANCH

002660                                        GETRV   Z.DAT(R3),#N$XLEN ; READ XPT DDB
002700  010067  000074G                       MOV     R0,N$ADJ2+$BFR    ; STORE SECOND ADJACENCY START
002704                                        PUTRC                     ; WRITE XPT DDB TO IMAGE

002710                                10$:    RETURN

002712                                101$:   EMSG$R  9D                ; POOL.. BLOCK ALLOCATION FAILURE

```

```
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE             REFERENCES

ALVBLK     000346 R        #8-337
AXSYM      000460 R         7-313       #9-374
A$LEN      000004           14-824
C$LEN      000012           13-714
DSPTAB     000024 R        #6-122        10-414      10-461     *10-468     *10-469      10-470
DTBLK      000114 R         6-163       #6-170
DTSYM      000120 R        #6-171       *10-473
DTVAL      000124 R        #6-172        10-482
D$END    = 000104           12-618       13-707      14-817      16-989      17-1096
D$LNUM     000014           14-822
D$MLL      000040           13-712
D$MNOD     000041           17-1098
D$NA       000062           12-624       16-995
D$NBEA     000056           14-821
D$NBRA     000054           14-820
D$NN       000060           12-619       12-622      13-708      14-818      16-990
ETAB1      000222 R        #6-206        10-506
ETAB2      000232 R        #6-214        7-322
E$NBR      000014          #5-68
E$NBS      000020          #5-68
E$NCR      000034          #5-68
E$NCS      000036          #5-68
E$NIC      000044          #5-68
E$NLEN     000050          #5-68         17-1101     17-1123     17-1124     17-1129     17-1136     17-1139
E$NLLA     000012          #5-68
E$NLNK     000000          #5-68
E$NML      000040          #5-68
E$NMR      000024          #5-68
E$NMS      000030          #5-68
E$NNOD     000002          #5-68
E$NRT      000042          #5-68
E$NRTP     000005          #5-68
E$NSEG     000010          #5-68
E$NTIM     000046          #5-68
E$NUSE     000004          #5-68
E$STRT     000006          #5-68
FNDPDV     004426 R         11-524       13-784     #18-1161
F.RSIZ   = ******   GX     *7-270       *7-308      *7-311      *8-351      *11-571     *11-577     *11-590     *12-618     *12-640
                           *12-681      *13-707     *13-720     *13-750     *13-774     *13-788     *14-817     *14-835     *14-838
                           *14-845      *15-877     *15-899     *15-917     *15-928     *15-942     *15-951     *16-989     *16-1010
                           *16-1053     *16-1065    *17-1096    *17-1111    *17-1123    *17-1141
I$$AS    = ******           5-76         5-76        7-289       7-290       12-633      15-936      16-1003     17-1105
LF.ACT   = 100000          #5-72
LF.BRO   = 000400          #5-72
LF.BWT   = 000007          #5-72
LF.ENA   = 002000          #5-72
LF.LPB   = 001000          #5-72
LF.MDC   = 000100          #5-72
LF.MFL   = 004000          #5-72
LF.MTP   = 000020          #5-72         11-531
LF.PAC   = 000200          #5-72
LF.RDY   = 040000          #5-72
```

```
VRES - VNP ROOT MACRO V05.03b Tuesday 17-Sep-85 16:15  Page 5
MACRO DEFINITIONS

     63                                          .SBTTL  MACRO DEFINITIONS
     64                                  ;
     65                                  ; MACRO LIBRARY CALLS
     66                                  ;
     67
     68                                          .MCALL  FDBDF$,FDOP$A,FDRC$A,FSRSZ$,GCLDF$,VNPDF$,ASR$
     69                                          .MCALL  SLTDF$,PDVDF$,QIOW$.DIR$,ERMSG$,ERBLK$,ERRPT$
     70                                          .MCALL  SAVRG,RESRG,DHBDF$
     71
     72                                          .ENABL LC               ; ENABLE LOWER CASE
     73
     74                                  ;
     75                                  ; GLOBAL LIBRARY SYMBOLS
     76                                  ;
     77
     78 000000                                   SLTDF$  <=>,<:>         ; DEFINE SLT OFFSETS GLOBALLY
     79 000000                                   PDVDF$  <=>,<:>         ; DEFINE PDV OFFSETS GLOBALLY
     80 000000                                   VNPDF$                  ; DEFINE VNP OFFSETS
     81 000000                                   DHBDF$                  ; DEFINE DECNET HOME BLOCK OFFSETS
     82
```

```
002032          .ROUTM:::.BLKW   1         ; ROUTING TIMER INTERAL
002034          ..SCSR:::.BLKW   1         ; SECONDARY CSR FROM REQUEST BLOCK
002036          $$HTIM:::.BLKW   48.       ; HELLO TIMER
002176          $$LTIM:::.BLKW   48.       ; LISTEN TIMER
002336          $$NODC:::.BLKW   1         ; MAXIMUM NUMBER OF NODE COUNTERS
002340          $TRIB:: .BLKW    1         ; TRIBUTARY NUMBER

002342          $PDVA:: .BLKW    1         ; PDV ADDRESS
002344          $NAME:: .BLKW    1         ; RAD50 PDV NAME
002346          $BIAS:: .BLKW    1         ; RELOCATION BIAS
002350          $$INC:: .BLKW    1         ; INCREMENT SIZE IN 32. WORD BLOCKS
002352          $$PAR:: .BLKW    2         ; RAD50 PARTITION NAME
002356          $$DEV:: .BLKW    1         ; ASCII DEVICE NAME
002360          $$TBL:: .BLKW    1         ; DEVICE DISPATCH TABLE ADDRESS
002362          $$LINX:::.BLKW   1         ; LOGICAL LINK COUNT
002364          $CTIM:: .BLKW    1         ; COUNTER TIMER VALUE (LINE PARAMETER ALSO)
002366          $X3DBS:::.BLKW   1         ; DEFAULT BLOCK SIZE
002370          $X3MBS:::.BLKW   1         ; MAXIMUM BLOCK SIZE
002372          $X3DWS:::.BLKW   1         ; DEFAULT WINDOW SIZE
002374          $X3MWS:::.BLKW   1         ; MAXIMUM WINDOW SIZE
002376          $X3CAL:::.BLKW   1         ; CALL TIMER
002400          $X3CLR:::.BLKW   1         ; CLEAR TIMER
002402          $X3RES:::.BLKW   1         ; RESET TIMER
002404          $X3RST:::.BLKW   1         ; RESTART TIMER
002406          $X3MCL:::.BLKW   1         ; MAXIMUM CLEAR VALUE
002410          $X3MRE:::.BLKW   1         ; MAXIMUM RESETS VALUE
002412          $X3MRS:::.BLKW   1         ; MAXIMUM RESTARTS VALUE
                ;
                ; LINE PARAMETERS
                ;
002414          $SLTA:: .BLKW    1         ; SLT ADDRESS
002416          $DLCPV:::.BLKW   1         ; DLC PDV ADDRESS
002420          $DDMPV:::.BLKW   1         ; DDM PDV ADDRESS
002422          $SLN::   .BLKW   1         ; SYSTEM LINE NUMBER
002424                   .BLKB   48.       ; MUX SLN'S
002504          $$CSR:: .BLKW    1         ; CSR ADDRESS
002506          $$PRI:: .BLKW    1         ; INTERRUPT PRIORITY
002510          $$VECT:::.BLKW   1         ; VECTOR ADDRESS
002512          $$MTP:: .BLKW    1         ; MULTIPOINT STATION COUNT
002514          $$SNUM:::.BLKB   48.       ; MULTIPOINT STATION NUMBERS
002574          $$APR:: .BLKB    48.       ; MULTIPOINT ACTIVE RATIO
002654          $$DPR:: .BLKW    1         ; MULTIPOINT DEAD RATIO
002656          $$NBRA:::.BLKW   1         ; BROADCAST ROUTER ADJACENCIES
002660          $$RPRI:::.BLKW   1         ; ROUTER PRIORITY
002662          $$MUX:: .BLKW    1         ; MUX UNIT COUNT
002664          $$SCSR:::.BLKW   16.       ; SECONDARY CSR TABLE
002724          $$DCHA:::.BLKW   2         ; DEVICE CHARACTERISTICS
002730                   .BLKW   48.*2     ; MUX CHARACTERISTICS
003230          $$PCHA:::.BLKW   1         ; PROTOCOL EMULATOR CHARACTERISTICS
003232                   .BLKW   48.       ;  MUX CHARACTERISTICS
003372          $$LSAD:::.BLKW   1         ; LOCAL PHYSICAL STATION ADDRESS
003374          $X2MWS:::.BLKW   1         ; MAXIMUM WINDOW SIZE
003376          $X2MBS:::.BLKW   1         ; MAXIMUM BLOCK SIZE
003400          $X2MRT:::.BLKW   1         ; MAXIMUM RETRANSMIT COUNT
003402          $X2RET:::.BLKW   1         ; RETRANSMIT TIMER
003404          $X2HLD:::.BLKW   1         ; HOLDBACK TIMER
                ;
```

```
   1140                                                  .SBTTL  $PUTAD - PUT RECORD TO SYSTEM IMAGE FILE
   1141                                         ;+
   1142                                                  **-$PUTAD
   1143                                         ;
   1144                                         ;        A RECORD UP TO 512. BYTES IS WRITTEN TO THE SYSTEM
   1145                                         ;        IMAGE FILE AT THE ADDRESS CORRESPONDING TO THE BIAS
   1146                                         ;        AND VIRTUAL ADDRESS OFFSET FROM THAT BIAS.
   1147                                         ;
   1148                                         ; INPUTS:
   1149                                         ;        $RBLCK  - BIAS OF BUFFER IN REAL MEMORY
   1150                                         ;        $RADDR  - VIRTUAL OFFSET FROM BIAS
   1151                                         ;        SYSFDB+F.RSIZ - NUMBER OF BYTES TO BE WRITTEN
   1152                                         ;
   1153                                         ; OUTPUTS:
   1154                                         ;        RECORD AT $BFR IS WRITTEN TO SYSTEM IMAGE.
   1155                                         ;-
   1156
   1157 004700                                  $PUTAD::
   1158                                                  .IF DF   R$$MPL                                                       ;[MP01]
   1159                                                  TST      $RBLCK          ; IF ZERO BIAS                               ;[MP01]
   1160                                                  BEQ      $PUTRC          ; THEN BRANCH (PUTRC ACCESES DSR)            ;[MP01]
   1161                                                  .ENDC                                                                 ;[MP01]
   1162
   1163 004700  004267  000000G                         JSR      R2,$SAVVR       ; SAVE R0,R1,AND R2
   1164 004704  012746  000000G                         MOV      #.PUT,-(SP)     ; PUT SUBROUTINE ADDRESS ON STACK
   1165 004710  000404                                  BR       60$             ; JUMP RIGHT IN ...
```

```
VRES       CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE               REFERENCES

FL.TOP  = 040000  G         #15-584
FL.WRN  = 100000  G          15-568    #15-585
FNERR     001364 R          #13-387     14-414
FNPTR     001362 R          *8-253     #13-386     14-415
F.DFNB  = 000046              6-102      6-113
F.DSPT  = 000044              6-102      6-113
F.ERR   = 000052             23-1287
F.HIBK  = 000004             19-1115
F.LUN   = 000042              6-102      6-113
F.NRBD  = 000024            *19-1120
F.RSIZ  = 000002              6-117     19-1120     19-1122
F.URBD  = 000020              6-102      6-102      6-102       6-102       6-112
GCLBUF    000030 RG           6-102     #6-104      6-105
GCLSIZ  = 000226              6-102     #6-105
G.MCRB  = 000002  G           6-102      6-102      6-102
IE.EOF  = ******  GX         22-1270
IO.WLB  = ******  GX          6-96
I$$AS   = ******              5-79      24-1320
LD.ALI  = 000030             #5-80
LD.CEX  = 000024             #5-80
LD.CIR  = 000035             #5-80
LD.LIN  = 000025             #5-80
LD.LOG  = 000031             #5-80
LD.MOD  = 000037             #5-80
LD.NOD  = 000027             #5-80
LD.OBJ  = 000032             #5-80
LD.PRO  = 000026             #5-80
LEVL1   = 000001  G         #15-642
LF.ACT  = 100000  G          #5-78
LF.BRO  = 000400  G          #5-78
LF.BWT  = 000007  G          #5-78
LF.ENA  = 002000  G          #5-78
LF.LPB  = 001000  G          #5-78
LF.MDC  = 000100  G          #5-78
LF.MFL  = 004000  G          #5-78
LF.MTP  = 000020  G          #5-78
LF.PAC  = 000200  G          #5-78
LF.RDY  = 040000  G          #5-78
LF.REA  = 010000  G          #5-78
LF.SER  = 000040  G          #5-78
LF.TIM  = 000010  G          #5-78
LF.UNL  = 020000  G          #5-78
LF.X2P  = 000000  G          #5-78
LINBUF    001000 RG          #7-205
LINSIZ    001024 RG          #7-210
LN.CLO  = 000000  G          #5-78
LN.DUM  = 000005  G          #5-78
LN.LOA  = 000004  G          #5-78
LN.LOO  = 000003  G          #5-78
LN.OAU  = 000003  G          #5-78
LN.OFF  = 000001  G          #5-78
LN.ON   = 000000  G          #5-78
```

```
                                                                  M 13
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16        PAGE 15
SYMBOL CROSS REFERENCE                                      CREF   04.00

SYMBOL   VALUE            REFERENCES

..C00      003554 RG      #16-695
..SCSR     002034 RG      #15-509
...PC1   = 000256 R       #6-102     6-102     6-102     6-102     6-102     6-102    #6-102     6-102     6-102
                           6-102     6-102     6-102    #6-102     6-102     6-102    #6-111     6-112     6-113
                           6-113     6-113     6-114     6-114    #6-114     6-114     6-114     6-114     6-114
                           6-114    #6-114
...PC2   = 000452 R       #6-102     6-102    #6-102     6-102    #6-102     6-102    #6-102     6-102     6-102
                          #6-102     6-102     6-102    #6-102     6-102    #6-102     6-102    #6-102     6-102
                          #6-112     6-112    #6-113     6-113    #6-113     6-113    #6-113     6-113    #6-114
                           6-114     6-114    #6-114     6-114     6-114    #6-114     6-114
...PC3   = 000256 R       #6-102     6-102    #6-114     6-114
```

N 13

```
                                         .SBTTL  CEXLD - CHECK COMM EXEC LOADED
                                 ;+
                                 ; CEXLD - MAKE SURE COMM EXEC IS LOADED
                                 ;
                                 ; INPUTS:
                                 ;        NONE
                                 ;
                                 ; OUTPUTS:
                                 ;        C-BIT=SUCCESS/FAILURE
                                 ;        DATA BLOCK ALLOCATED IF NECCESSARY FOR 11D/IAS
                                 ;-
000176                           CEXLD:
000176  032767  000000G 000000G          BIT     #FE.CEX,$FMASK  ; IS COMM EXEC LOADED?
000204  001401                           BEQ     101$            ; IF EQ, NO
000206                           10$:    RETURN
                                 ;
                                 ; ERROR CONDITION
                                 ;
000210                           101$:   EMSG$R  98              ; COMM EXEC NOT LOADED
```

```
                                                .SBTTL  SLIN - CHECK SET LINE PARMS
                                        ;+
                                        ; SLIN - CHECK "SET LINE" REQUEST BLOCK
                                        ;
                                        ; INPUTS:
                                        ;       $SLTA=SLT ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       R4 CLEARED IF NO WORK TO DO
                                        ;-
001544  012700  000346'                 SLIN:   MOV     #LBITS,R0       ; DO BIT TWIDDLING
001550                                          CALL    $BITS           ;
001554  016700  000000G                         MOV     $SLTA,R0        ; GET SLT ADDRESS
001560  032760  000000G 000000G                 BIT     #LF.RDY,L.FLG(R0) ; IS LINE READY ?
001566  001412                                  BEQ     10$             ; NO ..
001570  005067  000000G                         CLR     $MISS           ; ELSE, NOTHING MISSING - NOW
001574  105767  000000G                         TSTB    $OPTON          ; WAS THERE A REQUEST TO LOAD THE LINE ?
001600  001416                                  BEQ     30$             ; IF EQ, NO .. OKAY
001602  005004                                  CLR     R4              ; YES .. NO WORK TO DO!
001604                                          EMSG$R  ZB              ; LINE IN WRONG STATE
001612  000411                                  BR      30$

001614  005760  000000G                 10$:    TST     L.DDS(R0)       ; IS THERE ALREADY A DDM LINE TABLE ?
001620  001403                                  BEQ     20$             ; IF EQ, NO
001622  042767  000000G 000000G                 BIC     #MS.PRI,$MISS   ; ELSE, DON'T NEED CONTROLLER STUFF
001630  052767  000200  000000G 20$:    BIS     #OP$ALL,$OPTON  ; MAKE SURE LINE GET'S LOADED !

001636                                  30$:    RETURN
```

```
                                   .SBTTL  REP7 - REPLACE TASK NAME
                            ;+
                            ; REP7 - REPLACE REQUESTING TASK'S NAME
                            ;
                            ; NOTE:
                            ;          THIS ROUTINE HAS BEEN CHANGED INTO A NOP
                            ;
                            ; INPUTS:
                            ;          NONE
                            ;
                            ; OUTPUTS:
                            ;          NONE
                            ;
                            ;-
003344                      REP7:      RETURN
```

```
                                                                 N 1
VNTI        CREATED BY  MACRO  ON 29-JUN-85 AT 02:50     PAGE 10
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

NAMBUF     000022 R         #6-118    *24-1337   24-1340    24-1346   *24-1356   *24-1357    24-1365    24-1366    24-1464
                             24-1476
NAME       000030 R         #6-119     68-3130   68-3136    69-3167    69-3168    69-3169
NAMED      000000 R         #6-114   *104-4646  104-4650   104-4661  *104-4667
NBEAJ      000106 R         #6-143    *24-1334   24-1386
NBRA       000102 R         #6-141      8-458   *60-2736
NN         000074 R         #6-138    *24-1328   24-1380    24-1398
NOAD       014354 RG        #91-3894
NOAR       014334 RG        #91-3888
NODC       004620 R         #25-1505
NODCT      010420 R         #67-3025
NONMAP   = ******   GX        8-411     11-690    11-698     11-710   *24-1395   *24-1411   *24-1425   *24-1450   *24-1458
                            *30-1750  *30-1809
NTIERR     001410 R           8-238      8-241     8-323      8-371      8-455      8-474      8-488      8-494      8-501
                              8-505     #9-541    11-750     11-[illegible]    11-752     30-1791    30-1806    30-1820    59-2698
                             60-2727    67-3023   67-3033    67-3042    67-3052    67-3061    67-3070    67-3079    67-3088
                             67-3098    67-3108   69-3198    79-3516    86-3702    93-4015    94-4029   103-4519   107-4792
                            108-4830   109-4864
NTPNAM   = ******   GX        8-248     *8-258    *8-259     24-1461    26-1556    26-1557
NTPPAR   = ******   GX       17-1048   *17-1050   17-1070    17-1072    24-1463
NTPSAV   = ******   GX      *26-1547    26-1622   26-1634
NTPSCN     005606 R           8-239    #30-1747
NTPSIZ   = ******   GX        8-479      8-491   *24-1375   *24-1378   *24-1396   *24-1412   *24-1426   *24-1439   *24-1451
                            *24-1488    26-1547   26-1548   *26-1548    26-1559    26-1560    26-1580    26-1601   *26-1622
                            *26-1634   *30-1748  *30-1749    30-1821   *30-1821    30-1826   *30-1826    30-1834   *30-1834
NTWARN     001554 R         #10-580
N$$VCT   = ******            65-2925    65-2955   69-3174    79-3505    80-3541    86-3685    87-3725    88-3761    94-4033
                             98-4271   102-4450  103-4523   103-4529   105-4706
OF$INS   = 000001           #5-78
OF$LOG   = 000100           #5-78
OF$OFF   = 000002           #5-78
OF$ON    = 000000           #5-78
OF$SMC   = 000200           #5-78
ONAME      000032 R         #6-120    *82-3581  *82-3582   *83-3607   *83-3611    86-3680    86-3681
OPNCFG     005560 R           8-236     28-1700  #29-1726
OP$ADD   = 000200           #5-78
OP$ALL   = 000200           #5-78     107-4778
OP$CCS   = 000004           #5-78
OP$CIE   = 004000           #5-78
OP$CON   = 000400           #5-78
OP$COP   = 000001           #5-78
OP$COS   = 000100           #5-78
OP$DST   = 000200           #5-78
OP$DTE   = 000040           #5-78
OP$DUP   = 000002           #5-78
OP$EST   = 000100           #5-78
OP$EVE   = 010000           #5-78
OP$FIL   = 000200           #5-78
OP$GRP   = 000100           #5-78
OP$HOS   = 000001           #5-78
OP$HTM   = 000002           #5-78
OP$INA   = 004000           #5-78
```

```
                                                           N  2
VNTI       CREATED BY  MACRO  ON 29-JUN-85 AT 02:50     PAGE 23

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME        REFERENCES

FLTDF$            #5-57     5-77
FNBDF$            #5-58     5-80
GCLDF$            #5-55     5-70
GCL$              #5-70     28-1702   30-1752
GETAD             #5-56     65-2926   80-3542   87-3726   88-3762   98-4272   103-4524  103-4530  105-4707
GETRC             #5-56
GETRV             #5-56     8-270     8-287     8-329     8-457     16-948    17-1054   26-1573   26-1577   26-1583
                   26-1594  26-1597   26-1607   26-1614   61-2753   62-2792   62-2808   63-2841   63-2863   64-2889
                   65-2920  67-3017   67-3027   67-3037   67-3046   67-3056   67-3065   67-3074   67-3083   67-3093
                   67-3103  67-3110   69-3184   70-3239   71-3257   72-3280   73-3303   74-3318   75-3344   75-3372
                   76-3392  76-3408   88-3747   94-4063   98-4258   102-4454
HWDDF$            #5-57     5-79
ISTAT$            #5-56     18-1108   19-1139   20-1178   #21-1220  #22-1250  #23-1265  32-1884   #46-2329
LLTDF$            #5-58     5-63
MTRAN$            #18-1108  #19-1139  #20-1178  #21-1220  #22-1250  #23-1265  #32-1884  #46-2329
NHWDF$            #5-55     5-75
NTIERR            #5-87     8-238     8-241     8-323     8-371     8-455     8-474     8-488     8-494     8-501
                   8-505    11-750    11-751    11-752    30-1791   30-1806   30-1820   59-2698   60-2727   67-3023
                   67-3033  67-3042   67-3052   67-3061   67-3070   67-3079   67-3088   67-3098   67-3108   69-3198
                   79-3516  86-3702   93-4015   94-4029   103-4519  107-4792  108-4830  109-4864
NTWARN            #5-92
OBJDF$            #5-57     5-61
OFF$              #10-624
OPTDF$            #5-57     5-65
PCBDF$            #5-55     5-74
PDVDF$            #5-55     5-72
PUTAD             #5-56     8-402     8-418     8-448     13-834    14-879    65-2956   69-3175   79-3504   86-3686
                   94-4060  102-4451  103-4526  103-4536  105-4709
PUTRC             #5-56     8-293     8-298     8-337     8-459     11-742    12-797    26-1569   26-1587   26-1592
                   26-1611  26-1616   61-2778   62-2801   62-2821   63-2844   63-2868   64-2894   65-2960   67-3020
                   67-3030  67-3039   67-3048   67-3058   67-3067   67-3076   67-3085   67-3095   67-3105   67-3112
                   69-3191  70-3243   71-3266   72-3291   73-3306   74-3330   75-3357   75-3376   76-3396   76-3420
                   94-4076  102-4467
QDPB$             #10-624   10-624
QIOW$             #5-55     10-624
RAND              #8-270    8-270     #8-287    8-287     #8-293    8-293     #8-298    8-298     #8-329    8-329
                  #8-337    8-337     #8-402    8-402     #8-418    8-418     #8-448    8-448     #8-457    8-457
                  #8-459    8-459     #11-742   11-742    #12-797   12-797    #13-834   13-834    #14-879   14-879
                  #16-948   16-948    #17-1054  17-1054   #26-1569  26-1569   #26-1573  26-1573   #26-1577  26-1577
                  #26-1583  26-1583   #26-1587  26-1587   #26-1592  26-1592   #26-1594  26-1594   #26-1597  26-1597
                  #26-1607  26-1607   #26-1611  26-1611   #26-1614  26-1614   #26-1616  26-1616   #61-2753  61-2753
                  #61-2778  61-2778   #62-2792  62-2792   #62-2801  62-2801   #62-2808  62-2808   #62-2821  62-2821
                  #63-2841  63-2841   #63-2844  63-2844   #63-2863  63-2863   #63-2868  63-2868   #64-2889  64-2889
                  #64-2894  64-2894   #65-2920  65-2920   #65-2926  65-2926   #65-2956  65-2956   #65-2960  65-2960
                  #67-3017  67-3017   #67-3020  67-3020   #67-3027  67-3027   #67-3030  67-3030   #67-3037  67-3037
                  #67-3039  67-3039   #67-3046  67-3046   #67-3048  67-3048   #67-3056  67-3056   #67-3058  67-3058
                  #67-3065  67-3065   #67-3067  67-3067   #67-3074  67-3074   #67-3076  67-3076   #67-3083  67-3083
                  #67-3085  67-3085   #67-3093  67-3093   #67-3095  67-3095   #67-3103  67-3103   #67-3105  67-3105
                  #67-3110  67-3110   #67-3112  67-3112   #69-3175  69-3175   #69-3184  69-3184   #69-3191  69-3191
                  #70-3239  70-3239   #70-3243  70-3243   #71-3257  71-3257   #71-3266  71-3266   #72-3280  72-3280
                  #72-3291  72-3291   #73-3303  73-3303   #73-3306  73-3306   #74-3318  74-3318   #74-3330  74-3330
                  #75-3344  75-3344   #75-3357  75-3357   #75-3372  75-3372   #75-3376  75-3376   #76-3392  76-3392
```

```

                                                .SBTTL  CR.MC - CREATE MICROCODE PARTITION
                                        ;+
                                        ; CR.MC - CREATE MICROCODE PARTITION
                                        ;
                                        ; INPUTS:
                                        ;       $NAME - RAD50 PROCESS NAME
                                        ;
                                        ; OUTPUTS:
                                        ;        MICROCODE PARTITION CREATED
                                        ;        XXXMC.TSK LOADED INTO PARTITION
                                        ;-

000220  032767  000000G 000000G CR.MC:  BIT     #VF.SS,$VFLAG     ; S SYSTEM ?
000226  001550                          BEQ     90$               ; IF NO - BRANCH

000230  016767  000000G 000000G         MOV     $PARHD,$BFR       ; COPY FIRST PCB ADDRESS
000236  016705  000000G         10$:    MOV     $BFR,R5           ; ...
000242  001001                          BNE     20$               ; IF NOT END OF LIST - BRANCH
000244  000420                          BR      30$               ; ELSE - NOT SERVER BASE

000246                          20$:    GETRV   R5,,PLGTH         ; READ PCB
000264  022767  055260  000004G         CMP     #^RNTP,P.NAM+$BFR ; MATCH ?
000272  001361                          BNE     10$               ; IF NO - BRANCH
000274  022767  060044  000006G         CMP     #^RODL,P.NAM+2+$BFR ; MATCH ?
000302  001355                          BNE     10$               ; IF NO - BRANCH
000304  000521                          BR      90$               ; IF SERVER BASE - BRANCH

000306  016705  000000G         30$:    MOV     $DDMPV,R5         ; GET DDM PDV ADDRESS
000312  026567  000000G 000000G         CMP     Z.NAM(R5),$NAME   ; ARE WE LOADING DDM ?
000320  001113                          BNE     90$               ; IF NO - BRANCH
000322  032767  000000G 000000G         BIT     #FL.LMC,$FLAGS    ; LOAD MICROCODE ?
000330  001507                          BEQ     90$               ; IF NO - BRANCH
000332  005767  000250'                 TST     ERROR             ; ERROR OCCUR ?
000336  001053                          BNE     70$               ; IF YES - BRANCH

000340  016767  000252' 000152'         MOV     PCBA,TMPPCB       ; SAVE PCB ADDRESS
000346  005067  000252'                 CLR     PCBA              ; INDICATE NO MC PARTITION YET

000352                                  CALL    SFIL              ; SET UP MICROCODE FILE SPEC
000356                                  CALL    $CLOPE            ; OPEN FILE
000362  103474                          BCS     100$              ; IF FAILURE - BRANCH

000364                                  CALL    RMCLBL            ; READ LABEL
000370  103500                          BCS     110$              ; IF FAILURE - BRANCH
000372  052767  000001  000150'         BIS     #MCLOA,MCFLG      ; SET MICROCODE PARTITION FLAG
000400                                  CALL    PR.PAR            ; CREATE PARTITION
000404  103402                          BCS     40$               ; IF FAILURE - BRANCH

000406                                  CALL    PR.XFR            ; TRANSFER FILE TO PARTITION

000412                          40$:    CALL    $CLDEQ            ; CLOSE FILE
000416  016701  000252'                 MOV     PCBA,R1           ; STORE MICROCODE PARTITION ADDRESS
000422  042767  000001  000150'         BIC     #MCLOA,MCFLG      ; INIT MICROCODE FLAG
000430  000241                          CLC                       ; NO ERROR
000432  005767  000250'                 TST     ERROR             ; ERROR ?
000436  001415                          BEQ     80$               ; IF ON - BRANCH
```

```
                                                .SBTTL  PR.XFR - READ PROCESS INTO CORE
                                        ;+
                                        ; PR.XFR - READ PROCESS TASK IMAGE INTO CORE
                                        ;
                                        ; THIS ROUTINE IS CALLED TO READ THE PROCESS TASK IMAGE INTO CORE.
                                        ; IT THEN DETERMINES IF THE PROCESS HAS FREE SPACE AND, IF SO,
                                        ; SETS UP THE FIRST 2 WORDS AT THE OFFSET AS:
                                        ;       0               NO NEXT SECTION
                                        ;       NUMBER OF BYTES OF FREE SPACE THIS SECTION
                                        ;
                                        ;
                                        ; INPUTS:
                                        ;       PCBA=PCB ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       C-BIT=SUCCESS/FAILURE
                                        ;       R0,R1,R2=DESTROYED
                                        ;-

003026                                  PR.XFR: .IF     DF,R$$MPL
                                                BIT     #FL.CEX,$FLAGS  ; LOADING THE COMM EXEC?
                                                BEQ     30$             ; BR IF NO
                                                MOV     .CEX,R0         ; COPY INTO ARGUMENT REGISTER
                                                ADD     #77,R0          ; ROUND UP TO NEXT BLOCK
                                                ASR$    6,R0            ; CONVERT ADDRESS TO BLOCK BIAS
                                                BIC     #176000,R0      ; CLEAR OUT PROPAGATED SIGN BITS
                                                BR      40$             ; AND DO THE TRANSFER
                                                .ENDC

003026                                  30$:    GETRV   PCBA,.PLGTH     ; READ IN PCB
003046  016700  000014G                         MOV     P.REL+$BFR,R0   ; GET PARTITION STARTING ADDRESS

003052                                  40$:    CALL    $READX          ; READ INTO EXTENDED MEMORY
003056  103401                                  BCS     101$            ; IF CS, FAILURE
003060                                          RETURN

                                        ;
                                        ; ERROR CONDITION
                                        ;
003062  116777  000000G 000000G         101$:   MOVB    $IOSB,@$CLIOS   ; COPY ERROR CODE
003070  012767  000042  000250'                 MOV     #ERR34,ERROR    ; PROCESS IMAGE READ FAILURE
003076                                          RETURN


```

```
     60                                              .SBTTL  MACROS AND SYMBOLS AND INITIALIZATION
     61                                              .ENABLE LC
     62                                      ;
     63                                      ; DECLARE EXTERNAL MACROS
     64                                      ;
     65                                              .MCALL  ISTAT$,PRMPT$
     66                                      ;
     67                                      ; SIGNAL PROMPTING SUPPORT
     68                                      ;
     69          000001                              $GPRM   = 1
     70                                      ;
     71                                      ; MAKE TABLES READ ONLY
     72                                      ;
     73          000001                              $RONLY  = 1
```

```
                                        ;+
                                        ; $XFER - TRANSFER PROCESS FROM DISK TO CORE
                                        ;
                                        ; INPUTS:
                                        ;       $PDVA=PDV ADDRESS
                                        ;       $NAME=RAD50 PDV NAME
                                        ;
                                        ; OUTPUTS:
                                        ;       C-BIT=SUCCESS/FAILURE
                                        ;       $BIAS=PROCESS RELOCATION BIAS
                                        ;-
000000                                  $XFER::
                                                .MCALL  VNPBPT
000000                                          VNPBPT  XFER
000000  016700  000000G                         MOV     $PDVA,R0        ; GET PDV ADDRESS
000004  005760  000000G                         TST     Z.PCB(R0)       ; IS PROCESS ALREADY CORE-RESIDENT?
000010  001147                                  BNE     60$             ; IF NE, YES - NETLDR LOGIC ERROR
000012  016700  000000G                         MOV     $NAME,R0        ; GET PDV NAME
000016                                          CALL    $DFN            ; SETUP DEFAULT FILE NAME
000022                                          CALL    PRSYM           ; LOOK FOR SPECIAL SYMBOLS
000026  103534                                  BCS     50$             ; IF CS, FAILURE
000030  005000                                  CLR     R0              ; SETUP ASCIZ FILE SPEC
000032  005001                                  CLR     R1              ; FOR XXX.TSK
000034  005002                                  CLR     R2              ;       ...
000036                                          CALL    $AZFS           ;       ...
000042                                          CALL    $PRIO           ; TRANSFER PROCESS INTO CORE
000046  103534                                  BCS     TSKERR          ; IF CS, ERROR

                                                .IF DF  R$$MPL
                                                TST     VECVAL          ; IS PROCESS VECTORED ?
                                                BEQ     5$              ; IF NO - BRANCH
                                                CALL    ALVBLK          ; IF YES - ALLOCATE VECTOR BLOCK
                                                .ENDC

000050  016700  000000G                 5$:     MOV     $PDVA,R0        ; GET PDV ADDRESS
000054  010160  000000G                         MOV     R1,Z.PCB(R0)    ; STORE PCB ADDRESS
000060                                          GETRV   R1,.PLGTH       ; READ IN PCB
000076  016701  000000G                         MOV     $LDSIZ,R1       ; GET LOAD SIZE OF PROCESS
000102  016702  000000C                 10$:    MOV     $BFR+P.BLKS,R2  ; GET PARTITION SIZE IN BLOCKS
000106  020102                                  CMP     R1,R2           ; LOAD SIZE SAME AS PARTITION SIZE ?
000110  001001                                  BNE     20$             ; IF NE, NO - THERE MUST BE SOME SPACE LEFT
000112  005001                                  CLR     R1              ; ELSE, SAY THERE IS NO SPACE
000114  016760  000042' 000000G 20$:    MOV     PVAL,Z.DSP(R0)  ; SET DISPATCH TABLE ADDRESS (ASSUME UNMAPPED)
000122  005067  000000G                         CLR     $BIAS           ; ASSUME UNMAPPED SYSTEM

000126  005767  000000G                         TST     .MGMGE          ; IS THIS A MAPPED SYSTEM ?
000132  100432                                  BMI     30$             ; IF MI, NO
                                        ;
                                        ; MAPPED SYSTEMS
                                        ;
000134  012746  000000G                         MOV     #P.REL,-(SP)    ; COMPUTE ADDRESS OF RELOCATION BIAS
000140  062716  000000G                         ADD     #$BFR,(SP)      ; ...
000144  017660  000000  000000G                 MOV     @(SP),Z.DSP(R0) ; SET PROCESS RELOCATION BIAS
000152  013667  000000G                         MOV     @(SP)+,$BIAS    ; TWICE
000156  016760  000042' 000002G                 MOV     PVAL,Z.DSP+2(R0) ; SET DISPATCH TABLE ADDRESS
000164                                          ASL$    6,R1            ; GET LOAD SIZE IN BYTES
000200                                          ASL$    6,R2            ; GET PARTITION SIZE IN BYTES
```

```
                                               .SBTTL  $CKT - ALLOCATE XPT CIRCUIT COUNTERS

                                        ;+
                                        ; $CKT - ALLOCATE XPT CIRCUIT COUNTERS, INIT NODE COUNTERS
                                        ;
                                        ; INPUTS:
                                        ;       $PDVA - XPT PDV ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       INIT XPT NODE COUNTER
                                        ;       XPT CIRCUIT COUNTERS ALLOCATED
                                        ;               N$TLC   - NUMBER OF CIRCUIT COUNTERS
                                        ;               N$TLC+2 - BIAS  (NON UMR POOL..)
                                        ;               N$TLC+4 - VIRTUAL ADDRESS
                                        ;
                                        ;               N$TNC   - SIZE OF NODE COUNTERS
                                        ;               N$TNC+2 - VIRTUAL ADDRESS (PREALLOCATED FROM XPT)
                                        ;-

002720  016702  000000G                 $CKT::  MOV     $PDVA,R2                ; GET XPT PDV ADDRESS
002724  016267  000000G 000016'                 MOV     Z.DSP(R2),XPTDSP        ; GET XPT BIAS
002732  016202  000000G                         MOV     Z.DAT(R2),R2            ; GET XPT DDB ADDRESS

002736                                          GETRV   R2,#N$XLEN              ; READ XPT DDB
002754  016700  000114G                         MOV     N$TNC+$BFR,R0           ; GET # OF WORDS IN TRANSPORT NODE CTRS
002760  016701  000116G                         MOV     N$TNC+$BFR+2,R1         ; GET ADDRESS OF TRANSPORT NODE CTRS
002764  016702  000016'                         MOV     XPTDSP,R2               ; STORE XPT S BIAS

002770  022701  140000                          CMP     #140000,R1              ; ARE NODE COUNTERS IN SECOND 4K BLOCK
002774  003003                                  BGT     5$                      ; IF NO - BRANCH
002776  062702  000200                          ADD     #200,R2                 ; IF YES - ADJUST BIAS BY 4KW
003002  000442                                  BR      20$                     ; CONTINUE

003004  010103                          5$:     MOV     R1,R3                   ; COPY NODE COUNTER ADDRESS
003006  010004                                  MOV     R0,R4                   ; COPY SIZE OF COUNTERS
003010  006304                                  ASL     R4                      ; CONVERT TO BYTES
003012  060403                                  ADD     R4,R3                   ; CALCULATE TOP OF NODE COUNTERS
003014  022703  140000                          CMP     #140000,R3              ; DOES XPT NODE CTRS CROSS 4KW BOUNDARY
003020  002033                                  BGE     20$                     ; IF NOT - BRANCH

003022  012704  000000G                         MOV     #$BFR,R4                ; GET BUFFER ADDRESS
003026  010005                                  MOV     R0,R5                   ; COPY NODE COUNTERS SIZE
003030  005024                          10$:    CLR     (R4)+                   ; INIT XPT NODE COUNTERS
003032                                          SOB     R5,10$                  ; IF MORE - BRANCH

003036                                  15$:    PUTAD   R1,#2,R2                ; WRITE TWO BYTES OF XPT COUNTERS
003060  005300                                  DEC     R0                      ; COUNT ONE LESS WORD
003062  001431                                  BEQ     30$                     ; IF DONE - BRANCH

003064  005201                                  INC     R1                      ; ADDRESS NEXT WORD
003066  005201                                  INC     R1                      ; ...
003070  022701  140000                          CMP     #140000,R1              ; NEXT 4KW YET ?
003074  003360                                  BGT     15$                     ; IF NO - BRANCH
003076  062702  000200                          ADD     #200,R2                 ; ADJUST BIAS TOO NEXT 4K BLOCK
003102  012701  120000                          MOV     #120000,R1              ; SET NEW VIRTUAL ADDRESS
003106  000753                                  BR      15$                     ; CONTINUE

```

```
                                                                  N  8
VPRO        CREATED BY  MACRO  ON 28-AUG-85 AT 20:05      PAGE 2

SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE             REFERENCES

LF.REA  = 010000           #5-72
LF.SER  = 000040           #5-72
LF.TIM  = 000010           #5-72
LF.UNL  = 020000           #5-72
LF.X2P  = 000000           #5-72
LN.CLO  = 000000           #5-72
LN.DUM  = 000005           #5-72
LN.LOA  = 000004           #5-72
LN.LOO  = 000003           #5-72
LN.OAU  = 000003           #5-72
LN.OFF  = 000001           #5-72
LN.ON   = 000000           #5-72
LN.OOP  = 000004           #5-72
LN.OPE  = 000001           #5-72
LN.REF  = 000002           #5-72
LN.SER  = 000002           #5-72
LN.STA  = 000017           #5-72
LN.SUB  = 000360           #5-72
LN.TRI  = 000006           #5-72
LO.CON    000006           #5-76
LO.CSR    000006           #5-76
LO.CTL    000014           #5-76
LO.LIN    000004           #5-76
LO.PAR    000010           #5-76
LO.PRI    000010           #5-76
LO.SCR    000014           #5-76
LO.STA    000015           #5-76
LO.VCT    000012           #5-76
LR.AST    000026           #5-76
LR.CIR    000002           #5-76
LR.CTL    000004           #5-76
LR.EFN    000004           #5-76
LR.LIN    000002           #5-76
LR.PRM    000030           #5-76
LR.PRO    000002           #5-76
LR.STS    000000           #5-76
LR.TCB    000002           #5-76
LR.UNT    000005           #5-76
LS.AST  = 100000           #5-76
LS.CEX  = 000001           #5-76
LS.CIR  = 000010           #5-76
LS.CXO  = 010000           #5-76
LS.CX5  = 000400           #5-76
LS.DLM  = 004000           #5-76
LS.ECH  = 001000           #5-76
LS.FDX  = 004000           #5-76
LS.HDX  = 010000           #5-76
LS.LIN  = 000003           #5-76
LS.LMC  = 000007           #5-76
LS.NTI  = 004000           #5-76
LS.ON   = 000011           #5-76
LS.OPT  = 000400           #5-76
```

```
    84                                          .SBTTL  DATA
    85                                  ;
    86                                  ; I/O DATA STRUCTURES
    87                                  ;
    88
    89 000000                                   FSRSZ$  2               ; ALLOW TWO FILES OPEN SIMULTANEOUSLY
    90                                                                  ;  SYSTEM IMAGE FILE AND GET COMCL
    91
    92                                  ;
    93                                  ; ERROR MESSAGE REPORTING
    94                                  ;
    95
    96 000000                           TIOUT:  QIOW$   IO.WLB,2,2,,,,<,,40>
    97
    98                                  ;
    99                                  ; GET COMMAND LINE DATA BASE
   100                                  ;
   101
   102 000030                                   GCLDF$  1,1,<VNP>,GCLBUF,GCLSIZ         ; GCL DATA BASE
   103                                          .GLOBL  CL.OPN          ; NEEDED TO LEAVE DATA FILES OPEN
   104 000030                           GCLBUF::.BLKB   150.            ; GET COMMAND LINE BUFFER
   105                 000226                   GCLSIZ=.-GCLBUF         ; BUFFER SIZE
   106
   107                                  ;
   108                                  ; SYSTEM IMAGE FILE VIRTUAL I/O DATA BASE
   109                                  ;
   110
   111 000256                           SYSFDB::FDBDF$                  ; SYSTEM IMAGE FILE FDB
   112 000416                                   FDRC$A  ,$BFR           ; BUFFFER ADDRESS
   113 000416                                   FDOP$A  3,$CLCSI+C.DSDS,SYSDNB ; LUN 3/DATASET DESCR. BLK./DEF. FNB
   114 000416                           SYSDNB: NMBLK$  RSX11M,SYS,,SY,0 ; DEFAULT FIELD VALUES
   115
   116 000454                           $RSIZ:: .BLKW   1               ; RECORD SIZE OF FILE WHEN OPENED
   117                 000260'          $RSIZE==SYSFDB+F.RSIZ           ; LENGTH OF RECORD TO READ/WRITE
   118 000456                           $RADDR::.BLKW   1               ; INPUT RECORD ADDRESS IN SYSTEM IMAGE
   119 000460                           $RBLCK::.BLKW   1               ; INPUT BLOCK NUMBER ( FOR $GETAD/$PUTAD )
   120 000462                           $VCTVB::.BLKW   1               ; VBN OF MEMORY ( 0-777 ) DISPLACED BY 'SAV'
   121 000464                           $VCTOF::.BLKW   1               ; BLOCK OFFSET OF VIRTUAL ZERO
   122 000466      000000               $XFRAD::.WORD   0               ; SYSTEM IMAGE TRANSFER ADDRESS
   123 000470      000000               $XFRSV::.WORD   0               ; SYSTEM IMAGE TRANSFER ADDRESS         ;[MP01]
   124 000472                           $LOWSZ::.BLKW   1               ; NUMBER OF BYTES DISPLACED BY SAVE
   125 000474      000000               $CPBIA::.WORD   0               ; CPU BIAS WORD
   126 000476      004120'              MAXOV:: $MAXOV                  ; NECESSARY FOR IMPACT MODULE
   127 000500      005622'              CEACC:: $CEACC                  ; CAN BE REMOVED AFTER MOD FILE IS FIXED N$$VCT
   128                                  ;
   129                                  ;       DATA STRUCTURES USED TO MANIPULATE ACCOUNTING BLOCKS IN
   130                                  ;       NTPOOL. ( UNLOAD BLOCKS, CSR BLOCKS, MICROCODE BLOCKS...)
   131                                  ;
   132
   133 000502      000000               $ADRZ:: .WORD   0               ; THESE TWO WORD ENABLE TRANSPARENT MANIPU-
   134 000504      000000               $ADRX:: .WORD   0               ; LATION OF 32.-WORD BLOCKS FROM $ALN18
   135 000506      000000               $UNBSZ::.WORD   0               ; RO VALURE TO GET 32. WORD BLOCK FROM $ALN18
   136 000510      000000               $UNBAD::.WORD   0               ; VIRTUAL ADDRESS OF CURRENT BLOCK
   137 000512                           $UNBLK::.BLKW   32.             ; BLOCK IMAGE BUFFER
   138 000612                           $UNEND::                        ; END OF LOCAL COPY OF UNLOAD BLOCK
   139
   140 000612                           $HMBUF::.BLKB   D$END           ; TEMPORARY DECNET HOME BLOCK BUFFER
```

```
                                ; NETLDR STATUS
                                ;
003406  000000                  $FLAGS::.WORD   0               ; FLAG WORD
003410  100000                          .WORD   FL.WRN          ; DEFAULT VALUES

        000001                  FL.DDM==1                       ; DDM BEING LOADED
        000002                  FL.DLC==2                       ; DLC BEING LOADED
        000004                  FL.LLC==4                       ; LLC BEING LOADED
        000010                  FL.CEX==10                      ; COMM EXEC BEING LOADED
        000020                  FL.CHA==20                      ; MUX CHARACTERISRICS UPDATE ONLY
        000040                  FL.ECH==40                      ; ECHO THE TEMPLATE
        000100                  FL.FDX==100                     ; FULL-DUPLEX
        000200                  FL.HDX==200                     ; HALF-DUPLEX
        000400                  FL.KMX==400                     ; COMIOP MULTIPLEXED LINE
        001000                  FL.LMC==1000                    ; MICRO-CODE LOAD NEEDED
        002000                  FL.MUX==2000                    ; MULTIPLEXED LINE
        004000                  FL.CNT==4000                    ; CONTROLLER ONLY SPECIFIED FOR LINE-ID
        010000                  FL.PX1==10000                   ; SUCCESSFUL PROCESS I/O TRANSFER
        020000                  FL.PX2==20000                   ; ...
        040000                  FL.TOP==40000                   ; LOAD FROM TOP OF PARTITION
        100000                  FL.WRN==100000                  ; WARNING MESSAGE CONTROL
                                ;
                                ; VNP STATUS FLAG
                                ;
003412                          $VFLAG::.BLKW   1

                                .IF DF  R$$MPL                                                  ;[MP01]
                                $DFLAG::.WORD   0               ; FLAG EXTENSION                ;[MP01]
                                .ENDC                                                           ;[MP01]

        000001                  VF.NTI==        1               ; NTINIT MODE
        000002                  VF.ON==         2               ; ON/OFF FUNCTION CODE MODIFIER
        000004                  VF.TRA==        4               ; TRANSMIT/ENABLE FUNCTION CODE MODIFIER
        000010                  VF.SS==         10              ; RSX11S SYSTEM IMAGE BEING PROCESSED
        000020                  VF.ECH==        20              ; SET CEX IN ECHO MODE
        000040                  VF.INI==        40              ; INITIALIZATION COMPLETED SUCCESSFULLY
        000100                  VF.FIX==        100             ; FIX NTINIT WHEN SETTING STATE LOCAL ON
        000200                  VF.VDT==        200             ; NOTIFY VDT TO OUTPUT BUFFERS
        000400                  VF.ADD==        400             ; NODE IDENTIFICATION IS AN ADDRESS
        001000                  VF.ACP==        1000            ; SET NODE STATE COMPLETED SUCCESSFULLY
        002000                  VF.PEM==        2000            ; PROTOCOL EMULATOR COMMAND
        004000                  VF.DLX==        4000            ; DLX ONLY MODE
        010000                  VF.SKA==        10000           ; SINK NODE ID FOR LOGGING IS AN ADDRESS
        020000                  VF.EAD==        20000           ; NODE ENTITY ID FOR LOGGING IS AN ADDRESS
        040000                  VF.PSI==        40000           ; SPECIFIED CIRCUIT IS A PSI CIRCUIT
        100000                  VF.SED==        100000          ; CIRCUIT SERVICE DISABLE

                                .IF DF  R$$MPL                                                  ;[MP01]
                                DF.DAS==        1               ; D-SPACE ENABLED               ;[MP01]
                                DF.SCO==        2               ; READ LOWER DISPLACEMENT POINTER ;[MP01]
                                DF.POL==        4               ; SECONDARY POOL SUPPORTED      ;[MP01]
                                DF.PDE==        10              ; PROCESS SPACE DEALLOCATION    ;[MP01]
                                DF.MXU==        20              ; MUX KRB ADDRESS ADJUSTMENT    ;[MP01]
                                .ENDC                                                           ;[MP01]

003414                          $VFLG1::.BLKW   1               ; ANOTHER FLAGS WORD
        000001                  V1.NSY==        1               ; FILE IS NOT A SYSTEM IMAGE FILE
```

```
                                         .SBTTL  $GETAD - GET RECORD FROM SYSTEM IMAGE FILE
                                ;+
                                ;        **-$GETAD
                                ;
                                ; THIS ROUTINE IS CALLED TO GET A RECORD FROM THE TASK IMAGE FILE.
                                ; THE VIRTUAL ADDRESS OF THE RECORD IS DEFINED BY THE CONTENTS OF
                                ; $RADDR AND $RBLCK, WHICH HOLD AN OFFSET FROM A 32. WORD BLOCK
                                ; AND A BLOCK NUMBER RESPECTIVELY (THAT IS, FORM AN ADDRESS DOUBLE
                                ; WORD).  THE SIZE OF THE RECORD IS DEFINED BY THE INPUT FILE FDB.
                                ;
                                ; INPUTS:
                                ;        $RBLCK  - BLOCK NUMBER (32. WORDS PER BLOCK) TO BE READ FROM.
                                ;        $RADDR  - OFFSET FROM BLOCK $RBLCK AT WHICH TO START READING.
                                ;                BITS 15, 14, 13 (USED IN A RUNNING SYSTEM TO
                                ;                SELECT A MAPPING REGISTER) ARE IGNORED.
                                ;        SYSFDB+F.RSIZ- NUMBER OF BYTES TO READ
                                ;
                                ; OUTPUTS:
                                ;
                                ;        THE SPECIFIED RECORD IS READ FROM THE TASK IMAGE
                                ;
                                ; NOTE: ALL REGISTERS ARE PRESERVED ACROSS CALL
                                ;-

 004712                         $GETAD::
                                         .IF DF  R$$MPL                                          ;[MP01]
                                         TST     $RBLCK                                          ;[MP01]
                                         BNE     61$                                             ;[MP01]
                                         CALLR   $GETRV                                          ;[MP01]
                                61$:                                                             ;[MP01]
                                         .ENDC                                                   ;[MP01]

 004712  004267  000000G                 JSR     R2,$SAVVR       ; SAVE R0, R1, R2
 004716  012746  000000G                 MOV     #.GET,-(SP)     ; SET ADDRESS OF RECORD ACCESS ROUTINE
 004722  010346                 60$:     MOV     R3,-(SP)        ; SAVE R3
 004724  012700  000256'                 MOV     #SYSFDB,R0      ; GET ADDRESS OF INPUT FILE FDB

                                ;
                                ;        THIS SECTION CONVERTS A DOUBLE WORD ADDRESS ( BLOCK,VIRTUAL
                                ;        ADDRESS ) USED BY MEMORY MANAGEMENTT INTO THE CORRESPONDING
                                ;        BLOCK AND OFFSET WITHIN THE SYSTEM IMAGE FILE.
                                ;
                                ;        SPECIAL CAUTION MUST BE TAKEN WHEN CALLING GETAD/PUTAD WITH
                                ;        ANY OF THE MEMORY MANAGEMENT BITS SET ( HIGH ORDER THREE BITS )
                                ;        SINCE THEY WILL BE TREATED AS PART OF THE VIRTUAL ADDRESS.
                                ;
 004730  016702  173524                  MOV     $RBLCK,R2       ; GET BLOCK NUMBER (UNITS OF 64. BYTES)
 004734  005003                          CLR     R3              ; CLEAR C-BIT
 004736  006002                          ROR     R2              ; CHANGE R2 TO 512. BYTE UNITS
 004740  106003                          RORB    R3              ; SAVE THREE LOWEST BITS FROM BLOCK
 004742  006202                          ASR     R2              ;  NUMBER (CONSTITUTE THREE HIGHEST
 004744  106003                          RORB    R3              ;  BITS OF OFFSET IN 512. BYTE BLOCK)
 004746  006202                          ASR     R2              ; ...
 004750  106003                          RORB    R3              ; ...
 004752  006303                          ASL     R3              ; ...
 004754  016746  173476                  MOV     $RADDR,-(SP)    ; GET ADDRESS
 004760  021627  120000                  CMP     (SP),#120000    ; IS ADDRESS MAPPED?
```

```
                                                                    N 12
VRES        CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 3
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

LN.OOP  = 000004   G        #5-78
LN.OPE  = 000001   G        #5-78
LN.REF  = 000002   G        #5-78
LN.SER  = 000002   G        #5-78
LN.STA  = 000017   G        #5-78
LN.SUB  = 000360   G        #5-78
LN.TRI  = 000006   G        #5-78
LO.ADD    000022            #5-80
LO.ADR    000052            #5-80
LO.CLS    000002            #5-80
LO.CON    000006            #5-80
LO.COP    000010            #5-80
LO.COS    000020            #5-80
LO.CSR    000010            #5-80        5-80
LO.CTL    000032            #5-80
LO.DES    000002            #5-80        5-80
LO.DTE    000047            #5-80
LO.EVT    000004            #5-80
LO.GRO    000024            #5-80
LO.HOS    000040            #5-80
LO.HTM    000032            #5-80
LO.INA    000062            #5-80
LO.INC    000060            #5-80
LO.LDT    000046            #5-80
LO.LID    000030            #5-80
LO.LIN    000004            #5-80
LO.LTM    000034            #5-80
LO.MAC    000022            #5-80
LO.MDE    000024            #5-80
LO.NAM    000044            #5-80
LO.NET  = 000002            #5-80
LO.NNM    000022            #5-80
LO.OFL    000012            #5-80
LO.ONM    000004            #5-80
LO.OUT    000056            #5-80
LO.OWN    000026            #5-80
LO.PAR    000010            #5-80
LO.PRI    000012            #5-80
LO.RET    000064            #5-80
LO.RPA    000030            #5-80
LO.SCR    000016            #5-80
LO.SEG    000054            #5-80
LO.SKA    000014            #5-80
LO.SKN    000014            #5-80
LO.SNM    000035            #5-80
LO.TPA    000020            #5-80
LO.TRB    000034            #5-80
LO.TRI    000030            #5-80
LO.UNT    000033            #5-80
LO.VCT    000014            #5-80
LO.VER    000042            #5-80
LR.ACC    000020            #5-80
```

```
                                                                        N 13
VRES       CREATED BY  MACRO  ON 17-SEP-85 AT 16:16      PAGE 16
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME        REFERENCES

ALUN$             #6-102     6-102
ASR$              #5-68      24-1320
BDOFF$            #6-89
CALL               8-255     9-279      9-282      10-318     11-342     11-345     11-353     14-436     14-440     19-1117
                   19-1124   19-1135    19-1137    19-1138    22-1254    22-1272    25-1348    25-1355
CBYTE$            #6-102     6-102      6-102     #6-102      6-102     #6-112      6-112     #6-113      6-113      6-113
CSI$              #6-102     6-102
CWORD$            #6-102     6-102      6-102      6-102     #6-102      6-102      6-102     #6-102      6-102      6-102
                  #6-112     6-112      6-112     #6-113      6-113      6-113      6-113     #6-114      6-114      6-114
DEF$L             #6-89
DHBDF$            #5-70      5-81
DIR$              #5-69     #6-102      14-420
ERBLK$            #5-69      23-1285    23-1287    23-1289    23-1292
ERMSG$            #5-69      23-1282    23-1283    23-1284    23-1286    23-1288    23-1290    23-1291
ERRPT$            #5-69      19-1135    19-1137    19-1138    22-1272
FDBDF$            #5-68     #6-102      6-102      6-111
FDBSZ$            #6-102     6-102     #6-111      6-111
FDOFF$            #6-102     6-102     #6-102      6-102     #6-112      6-112     #6-113      6-113
FDOP$A            #5-68     #6-102      6-102      6-113
FDRC$A            #5-68     #6-102      6-102      6-112
FSRSZ$            #5-68      6-89
GCLDF$            #5-68      6-102
GCL$              #6-102     22-1254
GMCR$             #6-102     6-102
HWDDF$            #18-997    18-998
NBOFF$            #6-102     6-102     #6-114      6-114
NMBLK$            #6-102     6-102      6-114
OFF$              #6-96     #6-102     #6-102     #6-102     #6-102     #6-102
PDVDF$            #5-69      5-79
QDPB$             #6-96      6-96      #6-102      6-102
QIOW$             #5-69      6-96
QIO$              #6-102     6-102
RAD50$            #6-102     6-102      6-102     #6-114      6-114      6-114
RESRG             #5-70      24-1324
RETURN             8-260     9-287      10-322     11-357     12-378     14-421     14-449     19-1126    19-1129    22-1264
                   24-1325
SAVRG             #5-70      24-1316
SETF$             #6-102     6-102
SLTDF$            #5-69      5-78
VNPDF$            #5-68      5-80
WTSE$             #6-102     6-102
```

```
                                        .SBTTL  PSILD - CHECK PSI LOADED
                                ;+
                                ; PSILD - MAKE SURE PSI IS LOADED
                                ;
                                ; INPUTS:
                                ;       NONE
                                ;
                                ; OUTPUTS:
                                ;       C-BIT=SUCCESS/FAILURE
                                ;
                                ;-
000216                          PSILD:
000216  005767  000000G                 TST     $PSIPT          ; IS PSI LOADED?
000222  001401                          BEQ     101$            ; IF EQ, NO
000224                          10$:    RETURN
                                ;
                                ; ERROR CONDITION
                                ;
000226                          101$:   EMSG$R  06              ; PSI NOT LOADED
```

```
                                                .SBTTL  SCIR - CHECK SET CIR PARMS
                                        ;+
                                        ; SCIR - CHECK "SET CIRCUIT" REQUEST BLOCK
                                        ;
                                        ; INPUTS:
                                        ;       $SLTA=SLT ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       NONE
                                        ;-
 001640 005767  000000G                 SCIR:   TST     $OPTON                  ; INITIAL CIRCUIT LOAD?
 001644 001411                                  BEQ     10$                     ; BR IF YES
 001646 016700  000000G                         MOV     $SLTA,R0                ; GET SLT ADDRESS
 001652 032760  000000G 000000G                 BIT     #LF.RDY,L.FLG(R0)       ; IS LINE READY ?
 001660 001003                                  BNE     10$                     ; BR IF YES
 001662                                         EMSG$R  ZB                      ; ELSE, LINE IN WRONG STATE
 001670                                 10$:    RETURN
```

```
B  1
C  1
D  1
E  1
F  1
G  1
H  1
I  1
J  1
K  1
L  1
M  1
N  1
B  2
C  2
D  2
E  2
F  2
G  2
H  2
I  2
J  2
K  2
L  2
M  2
N  2
B  3
C  3
D  3   VPIO
E  3
F  3
G  3
H  3
I  3
J  3
K  3
L  3
M  3
N  3
B  4
C  4
D  4
E  4
F  4
G  4
H  4
I  4
J  4
K  4
L  4
M  4
N  4
B  5
C  5
D  5
E  5
F  5
G  5
H  5
I  5
```

```
J  5   VPRMPT
K  5
L  5
M  5
N  5
B  6
C  6
D  6
E  6
F  6
G  6   VPRO
H  6
I  6
J  6
K  6
L  6
M  6
N  6
B  7
C  7
D  7
E  7
F  7
G  7
H  7
I  7
J  7
K  7
L  7
M  7
N  7
B  8
C  8
D  8
E  8
F  8
G  8
H  8
I  8
J  8
K  8
L  8
M  8
N  8
B  9
C  9
D  9
E  9
F  9
G  9
H  9
I  9   VRES
J  9
K  9
L  9
M  9
N  9
B 10
C 10
D 10
```

```
E 10
F 10
G 10
H 10
I 10
J 10
K 10
L 10
M 10
N 10
B 11
C 11
D 11
E 11
F 11
G 11
H 11
I 11
J 11
K 11
L 11
M 11
N 11
B 12
C 12
D 12
E 12
F 12
G 12
H 12
I 12
J 12
K 12
L 12
M 12
N 12
B 13
C 13
D 13
E 13
F 13
G 13
H 13
I 13
J 13
K 13
L 13
M 13
N 13
B 14   VRQB
C 14
D 14
E 14
F 14
G 14
H 14
I 14
J 14
K 14
L 14
```

```
M 14
N 14
B 15
C 15
D 15
E 15
F 15
G 15
H 15
I 15
J 15
K 15
L 15
M 15
N 15
B 16
C 16
D 16
E 16
F 16
G 16
H 16
I 16
J 16
K 16
L 16
M 16
```

```
                                        .SBTTL  REP7A - REPLACE TASK NAME AND CHANNEL NUMBER
                                ;+
                                ; REP7A - REPLACE REQUESTING TASK'S NAME AND CHANNEL NUMBER
                                ;
                                ; INPUTS:
                                ;       $$TASK=REQUESTING TASK'S NAME
                                ;       R1=MAPPING TABLE (WITHIN PDV) ADDRESS
                                ;       R3=FORMAT STRING ADDRESS
                                ;       R4=ERROR MESSAGE BUFFER ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       R0,R1,R2=DESTROYED
                                ;       R3,R4=UPDATED
                                ;-
003346  010146                  REP7A:  MOV     R1,-(SP)        ; SAVE CHANNEL TABLE POINTER
003350                                  CALL    REP7            ; REPLACE REQUESTING TASK'S NAME
003354  112314                  10$:    MOVB    (R3)+,(R4)      ; LOOK FOR THE SECOND ASTERISK
003356  122724  000052                  CMPB    #'*,(R4)+       ;       ...
003362  001374                          BNE     10$             ;       ...
003364  010400                          MOV     R4,R0           ; COPY BUFFER ADDRESS
003366  112740  000060                  MOVB    #'0,-(R0)       ; SKIP THE ASTERISK
003372  016701  000000G                 MOV     $PDVA,R1        ; GET PDV ADDRESS
003376  062701  000002G                 ADD     #Z.MAP+2,R1     ; POINT AT MAPPING TABLE
003402  162601                          SUB     (SP)+,R1        ; LESS INPUT VALUE
003404  001406                          BEQ     20$             ; IF EQ, NO WORK TO DO
003406  005401                          NEG     R1              ; CONVERT BACK TO A POSITIVE NUMBER
003410  006201                          ASR     R1              ; AND TO A BYTE INDEX
003412  005002                          CLR     R2              ; SUPPRESS LEADING ZEROES
003414                                  CALL    $CBOMG          ; CONVERT TO OCTAL
003420  010004                          MOV     R0,R4           ; SET NEW BUFFER ADDRESS
003422                          20$:    RETURN
```

```
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 7
SYMBOL CROSS REFERENCE                                        CREF   C4.00

SYMBOL  VALUE              REFERENCES

$C5TA   = ******  GX        34-1263
$DDMPV  = ******  GX       *19-596     19-631
$DEA16  = ******  GX        35-1311
$DEVHD  = ******  GX        24-857
$DLCPV  = ******  GX       *19-634
$ERR01    002036 R         #8-232      14-447
$ERR02    002074 R         #8-233      14-448
$ERR03    002134 R         #8-234      14-449
$ERR04    002210 R         #8-235      14-450
$ERR05    002252 R         #8-236      14-451
$ERR06    002320 R         #8-237      13-379
$ERRZB    001700 R         #8-229      22-751      23-776      24-820
$ERRZP    001734 R         #8-230      20-714
$ERRZR    001772 R         #8-231      24-805
$ERRZ1    001410 R         #8-222
$ERRZ3    001440 R         #8-223      24-850
$ERRZ4    001466 R         #8-224      24-877
$ERRZ5    001530 R         #8-225      24-903
$ERRZ6    001550 R         #8-226      25-934
$ERRZ7    001604 R         #8-227
$ERRZ8    001644 R         #8-228      26-975
$ERR86    001350 R         #8-221      19-643
$ERR87    001302 R         #8-220      29-1123
$ERR88    001220 R         #8-219      19-642
$ERR90    001164 R         #8-218      19-641
$ERR91    001126 R         #8-217      11-323
$ERR92    001064 R         #8-216      20-713
$ERR93    001020 R         #8-215      20-712
$ERR94    000764 R         #8-214      20-711
$ERR95    000730 R         #8-213      19-640
$ERR96    000652 R         #8-212      15-483
$ERR97    000614 R         #8-211      15-482
$ERR98    000540 R         #8-210      12-360
$ERR99    000506 R         #8-209      9-294
$FLAGS  = ******  GX        9-279     *9-279      *9-280     *11-318     *15-476      16-510     *19-600     *19-604     *19-608
                            24-807     24-889      27-1023     29-1107
$FMASK  = ******  GX        11-319     11-337      12-354      26-948
$FPDV     002776 R          14-399     15-470      19-592      26-957     #28-1046
$FSLT     003054 RG         19-618    #29-1081
$FUNC   = ******  GX       *9-286
$GETRV  = ******  GX        24-863     24-870      26-969      26-996      26-1003     35-1303     35-1313
$LLCTA  = ******  GX        14-414
$MISS   = ******  GX       *9-281     *16-502     *16-508     *16-514     *17-538     *20-668     *20-678     *20-684     *20-693
                           *20-701    *22-747     *22-756
$MXVCT  = ******  GX        20-691
$NAME   = ******  GX       *15-469    *19-591      24-803      24-811
$NLRQB    000000 RG        #9-267
$OPTON  = ******  GX        14-396     14-404      18-565      22-748     *22-757      23-771
$PDVA   = ******  GX       *15-472     17-539      24-796      24-856      32-1199
$PDVNM  = ******  GX        28-1049
$PDVTA  = ******  GX        19-626     28-1048     28-1061     34-1259
$PSIPT  = ******  GX        13-373
```

```
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 8
```

VSTI
Tab

VST

Table of contents

```
     5-   56     MACROS AND SYMBOLS AND INITIALIZATION
     6-  103     KEYWORDS
     7-  222     COMMANDS
     8-  244     CLEAR COMMAND
     9-  349     SET COMMAND
    10-  517     SHOW COMMAND
    11-  614     ENTITY ARGUMENT SUBEXPRESSIONS
    12-  777     ENTITY IDENTIFICATION SUBEXPRESSIONS
    13-  956     ENTITY OPTION SUBEXPRESSIONS
    14- 1456     OPTION KEYWORD AND VALUE SUBEXPRESSIONS
    15- 1542     UTILITY SUBEXPRESSIONS
    17- 1589     INTERNAL ACTION MODULES
    18- 1607     STMAT3, STMAT5 - SET MINIMUM CHARACTER MATCH
    19- 1628     PEMSTA - ACTION ROUTINE FOR THE PROTOCOL EMULATOR STATE
    574 000000                                  TRAN$  $LAMDA,         SHOXSV,     $IFX5S
    575 000000                                  TRAN$  $LAMDA,         SHOXSV,     $IFX9S
    576 000000                                  TRAN$  $LAMDA,         DONE
    577                                ;
    578                                ;      SHOW LOGGING OPTIONS
    579                                ;
    580 000000                                 STATE$  SHOLOG
    581 000000                                  TRAN$  !SHOTYP,        SHOLOG
    582 000000                                  TRAN$  !OPSINK,        SHOLOG,     ,           OP$SIN,         $OPTON
    583 000000                                  TRAN$  $LAMDA,         DONE
    584                                ;
    585                                ;      SHOW ALIAS OPTIONS
    586                                ;
    587 000000                                 STATE$  SHOALI
    588 000000                                  TRAN$  !SHOTYP,        SHOALI
    589 000000                                  TRAN$  !OPSCO,         SHOALI,     ,           OP$SCO,         $OPTON
    590 000000                                  TRAN$  $LAMDA,         DONE
    591                                ;
    592                                ;      SHOW MODULE X25-ACCESS OPTIONS
    593                                ;
    594 000000                                 STATE$  SHOXAC
    595 000000                                  TRAN$  !SHOTYP,        SHOXAC
    596 000000                                  TRAN$  !OPADS,         SHOXAC
    597 000000                                  TRAN$  $LAMDA,         DONE
    598                                ;
    599                                ;      SHOW MODULE X25-PROTOCOL OPTIONS
    600                                ;
    601 000000                                 STATE$  SHOXPR
    602 000000                                  TRAN$  !SHOTYP,        SHOXPR
    603 000000                                  TRAN$  !OPPDT,         SHOXPR
    604 000000                                  TRAN$  !OPGRO,         SHOXPR
    605 000000                                  TRAN$  $LAMDA,         DONE
    606                                ;
    607                                ;      SHOW MODULE X2n-SERVER OPTIONS
    608                                ;
    609 000000                                 STATE$  SHOXSV
    610 000000                                  TRAN$  !SHOTYP,        SHOXSV
    611 000000                                  TRAN$  !OPDST,         SHOXSV
    612 000000                                  TRAN$  $LAMDA,         DONE
```

```
   1241                                         ;
   1242                                         ; PRIORITY
   1243                                         ;
   1244 000000                                          STATE$  OPPRI
   1245 000000                                           TRAN$  &K$PRI
   1246 000000                                          STATE$  ,&P$P
   1247 000000                                           TRAN$  $DNUMB,          $EXIT,     $PRI,      LS.OPT,       RQBST
   1248                                         ;
   1249                                         ; RECEIVE PASSWORD
   1250                                         ;
   1251 000000                                          STATE$  OPRPAS
   1252 000000                                           TRAN$  !KWRPAS
   1253 000000                                          STATE$  ,&P$TPAS
   1254 000000                                           TRAN$  $STRNG,          $EXIT,     $STRPA
   1255                                         ;
   1256                                         ; SCOPE
   1257                                         ;
   1258 000000                                          STATE$  OPSCO
   1259 000000                                           TRAN$  &K$SCO,          OPSCO1
   1260 000000                                           TRAN$  $LAMDA,          OPSCO2
   1261 000000                                          STATE$  OPSCO1,&P$SCO
   1262 000000                                           TRAN$  $LAMDA
   1263 000000                                          STATE$  OPSCO2
   1264 000000                                           TRAN$  &K$GLO,          $EXIT
   1265 000000                                           TRAN$  &K$TER
   1266 000000                                          STATE$  ,&P$TER
   1267 000000                                           TRAN$  $ALPHA,          ,          $SCOP1
   1268 000000                                          STATE$
   1269 000000                                           TRAN$  $ALPHA,          ,          $SCOP2
   1270 000000                                          STATE$
   1271 000000                                           TRAN$  $NUMBR,          ,          $SCOP3
   1272 000000                                          STATE$
   1273 000000                                           TRAN$  <':>,            $EXIT,     $SCOND
   1274 000000                                           TRAN$  $LAMDA,          $EXIT,     $SCOND
   1275
   1276
   1277                                         ;
   1278                                         ; SERVICE
   1279                                         ;
   1280 000000                                          STATE$  OPSER
   1281 000000                                           TRAN$  &K$SER
   1282 000000                                          STATE$
   1283 000000                                           TRAN$  &K$ENA,          $EXIT
   1284 000000                                           TRAN$  &K$DIS,          $EXIT,     ,     VF.SED,   $VFLAG
   1285
   1286
   1287                                         ;
   1288                                         ; SINK NAME (FOR CLEAR LOGGING)
   1289                                         ;
   1290 000000                                          STATE$  OPCSNM
   1291 000000                                           TRAN$  &K$NAM,          $EXIT
   1292                                         ;
   1293                                         ; SINK NAME (FOR SET LOGGING)
   1294                                         ;
   1295 000000                                          STATE$  OPSNAM
   1296 000000                                           TRAN$  &K$NAM
   1297 000000                                          STATE$  ,&P$SNAM
```

```
                                       ;
                                       ; CHECK AHEAD IN INPUT LINE FOR BLANK TO END STRING
                                       ;
000134                                 STR.B:
000134   005703                                 TST     R3                      ;CHECK INPUT LENGTH
000136   003406                                 BLE     10$                     ;BR IF INPUT LENGTH <= 0
000140   121427   000040                        CMPB    (R4),#<' >              ;COMPARE NEXT CHARACTER TO ASCII BLANK
000144   001405                                 BEQ     20$                     ;BR IF CHARACTER = ASCII BLANK
000146   121427   000011                        CMPB    (R4),#TAB               ;COMPARE NEXT CHARACTER TO ASCII TAB
000152   001402                                 BEQ     20$                     ;BR IF CHARACTER = ASCII TAB
000154   062716   000002               10$:     ADD     #2,(SP)                 ;REJECT TRANSITION
000160                                 20$:     RETURN

         000001                                 .END
```

```
                                                        B  7
VSTB      CREATED BY   ACRO   ON 29-JUN-85 AT 03:03     PAGE 10
MACRO CROSS REFERENCE                                   CREF   04.00
MACRO NAME     REFERENCES

               #12-932    #12-934    #12-937    #12-943    #12-948    #12-953     13-968    #13-971    #13-973    #13-978
               #13-980    #13-985    #13-987    #13-993    #13-995    #13-1000   #13-1002   #13-1009   #13-1015   #13-1019
               #13-1026   #13-1029   #13-1031   #13-1036   #13-1039   #13-1041   #13-1047   #13-1049   #13-1055   #13-1058
               #13-1060   #13-1065   #13-1068   #13-1070   #13-1075   #13-1077   #13-1082   #13-1084   #13-1086   #13-1089
               #13-1091   #13-1098   #13-1100   #13-1105   #13-1107   #13-1113   #13-1115   #13-1120   #13-1124   #13-1130
               #13-1132   #13-1138   #13-1140   #13-1146   #13-1148   #13-1150   #13-1155   #13-1157   #13-1159   #13-1164
               #13-1166   #13-1168   #13-1173   #13-1175   #13-1177   #13-1182   #13-1184   #13-1186   #13-1189   #13-1191
               #13-1197   #13-1199   #13-1204   #13-1206   #13-1211   #13-1213   #13-1216   #13-1223   #13-1225   #13-1230
               #13-1232   #13-1237   #13-1239   #13-1244   #13-1246   #13-1251   #13-1253   #13-1258   #13-1261   #13-1263
               #13-1266   #13-1268   #13-1270   #13-1272   #13-1280   #13-1282   #13-1290   #13-1295   #13-1297   #13-1302
               #13-1305   #13-1308   #13-1316   #13-1318   #13-1325   #13-1327   #13-1329   #13-1331   #13-1338   #13-1340
               #13-1342   #13-1350   #13-1352   #13-1354   #13-1361   #13-1363   #13-1365   #13-1372   #13-1374   #13-1376
               #13-1383   #13-1386   #13-1389   #13-1394   #13-1399   #13-1401   #13-1406   #13-1408   #13-1413   #13-1416
               #13-1418   #13-1423   #13-1425   #13-1431   #13-1433   #13-1438   #13-1440   #13-1447   #13-1449   #13-1452
                14-1467   #14-1469   #14-1474   #14-1479   #14-1484   #14-1486   #14-1492   #14-1494   #14-1502   #14-1504
               #14-1506   #14-1508   #14-1511   #14-1513   #14-1517   #14-1519   #14-1524   #14-1530   #14-1532   #14-1534
               #14-1539    15-1556   #15-1564   #15-1567   #15-1572   #15-1574   #15-1576   #15-1579   #16-1587
 TRAN$         #5-61      #7-232     #7-233     #7-234     #7-235     #7-236     #7-237     #7-242     #8-254     #8-255
               #8-256     #8-258     #8-259     #8-260     #8-261     #8-263     #8-264     #8-265     #8-266     #8-267
               #8-268     #8-269     #8-276     #8-277     #8-282     #8-283     #8-284     #8-290     #8-291     #8-298
               #8-299     #8-305     #8-308     #8-309     #8-310     #8-311     #8-312     #8-313     #8-314     #8-319
               #8-324     #8-325     #8-326     #8-327     #8-328     #8-329     #8-334     #8-335     #8-340     #8-341
               #8-346     #8-347     #9-359     #9-361     #9-362     #9-363     #9-364     #9-366     #9-367     #9-368
               #9-369     #9-370     #9-371     #9-372     #9-377     #9-378     #9-381     #9-384     #9-385     #9-391
               #9-393     #9-394     #9-395     #9-396     #9-397     #9-398     #9-399     #9-400     #9-401     #9-408
               #9-409     #9-414     #9-415     #9-416     #9-417     #9-418     #9-419     #9-420     #9-421     #9-427
               #9-432     #9-433     #9-434     #9-435     #9-436     #9-437     #9-438     #9-439     #9-444     #9-449
               #9-450     #9-451     #9-452     #9-453     #9-454     #9-455     #9-456     #9-457     #9-458     #9-459
               #9-460     #9-461     #9-462     #9-467     #9-472     #9-473     #9-478     #9-481     #9-482     #9-487
               #9-488     #9-489     #9-490     #9-495     #9-496     #9-499     #9-500     #9-505     #9-506     #9-507
               #9-508     #9-513     #9-514     #9-515     #10-522    #10-523    #10-526    #10-527    #10-530    #10-531
               #10-536    #10-537    #10-538    #10-539    #10-540    #10-541    #10-542    #10-543    #10-544    #10-545
               #10-546    #10-547    #10-548    #10-549    #10-550    #10-551    #10-552    #10-553    #10-554    #10-555
               #10-556    #10-557    #10-562    #10-563    #10-564    #10-565    #10-570    #10-571    #10-572    #10-573
               #10-574    #10-575    #10-576    #10-581    #10-582    #10-583    #10-588    #10-589    #10-590    #10-595
               #10-596    #10-597    #10-602    #10-603    #10-604    #10-605    #10-610    #10-611    #10-612    #11-626
               #11-628    #11-633    #11-635    #11-640    #11-642    #11-647    #11-649    #11-654    #11-656    #11-661
               #11-663    #11-668    #11-670    #11-671    #11-676    #11-678    #11-683    #11-685    #11-690    #11-692
               #11-697    #11-699    #11-704    #11-706    #11-711    #11-713    #11-718    #11-720    #11-725    #11-727
               #11-732    #11-734    #11-739    #11-741    #11-746    #11-748    #11-753    #11-754    #11-756    #11-761
               #11-763    #11-768    #11-770    #11-775    #12-789    #12-791    #12-793    #12-794    #12-796    #12-801
               #12-803    #12-805    #12-806    #12-808    #12-809    #12-810    #12-812    #12-813    #12-815    #12-816
               #12-818    #12-819    #12-820    #12-825    #12-830    #12-831    #12-833    #12-835    #12-836    #12-838
               #12-840    #12-841    #12-843    #12-845    #12-846    #12-848    #12-849    #12-854    #12-856    #12-858
               #12-859    #12-861    #12-862    #12-864    #12-865    #12-870    #12-871    #12-872    #12-877    #12-878
               #12-881    #12-883    #12-884    #12-885    #12-888    #12-889    #12-891    #12-892    #12-897    #12-899
               #12-900    #12-903    #12-904    #12-905    #12-908    #12-915    #12-917    #12-918    #12-920    #12-921
               #12-922    #12-924    #12-926    #12-933    #12-935    #12-936    #12-938    #12-939    #12-944    #12-949
               #12-954    #13-969    #13-970    #13-972    #13-974    #13-979    #13-981    #13-986    #13-988    #13-989
               #13-994    #13-996    #13-1001   #13-1003   #13-1010   #13-1011   #13-1012   #13-1013   #13-1016   #13-1017
               #13-1020   #13-1027   #13-1028   #13-1030   #13-1032   #13-1037   #13-1038   #13-1040   #13-1042   #13-1048
               #13-1050   #13-1051   #13-1056   #13-1057   #13-1059   #13-1061   #13-1066   #13-1067   #13-1069   #13-1071
```

```
                                                        C  7
VSTB      CREATED BY  MACRO   ON 29-JUN-85 AT 03:03     PAGE 11
```

```
                                                                  B  8
VSYM       CREATED BY  MACRO  ON 29-JUN-85 AT 03:07       PAGE 2
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE                 REFFRENCES

L.NSTA     000014  G          #5-65
L.OWNR     000021  G          #5-65
L.UNT      000013  G          #5-65
MPAR     = 172100  G          #5-63
MPCSR    = 177746  G          #5-63
M$$MGE   = 000000              5-63
PIRQ     = 177772  G          #5-63
PMODE    = 030000  G          #5-63
PR0      = 000000  G          #5-63
PR1      = 000040  G          #5-63
PR2      = 000100  G          #5-63
PR3      = 000140  G          #5-63
PR4      = 000200  G          #5-63
PR5      = 000240  G          #5-63
PR6      = 000300  G          #5-63
PR7      = 000340  G          #5-63
PS       = 177776  G          #5-63
R$$MPL   = ******              5-63          5-63          5-63
R$$11D   = ******              5-63          5-64
R$$11M   = 000000              5-64
R$$11S   = ******              5-64
SF.ACT   = 000200  G          #5-65
SF.ENA   = 000100  G          #5-65
SF.LPB   = 000004  G          #5-65
SF.MFL   = 000040  G          #5-65
SF.PAC   = 000020  G          #5-65
SF.REA   = 000010  G          #5-65
SF.SER   = 000001  G          #5-65
SF.SVC   = 000002  G          #5-65
SF.UNL   = 000040  G          #5-65
SWR      = 177570  G          #5-63
S.COST     000001  G          #5-65
S.FLG      000000  G          #5-65
S.LEN      000004  G          #5-65
S.NMST     000002  G          #5-65
S.OWNR     000003  G          #5-65
TPS      = 177564  G          #5-63
UBMPR    = 170200  G          #5-63
UISAR0   = 177640  G          #5-63
UISAR1   = 177642  G          #5-63
X$$MCB   = ******              5-64          5-64
ZF.COU   = 001000  G          #5-64
ZF.DDM   = 000001  G          #5-64
ZF.DIA   = 004000  G          #5-64
ZF.DLC   = 000002  G          #5-64
ZF.DVP   = 100000  G          #5-64
ZF.INI   = 040000  G          #5-64
ZF.KMX   = 000020  G          #5-64
ZF.LLC   = 000004  G          #5-64
ZF.LMC   = 000100  G          #5-64
ZF.MAN   = 020000  G          #5-64
ZF.MFL   = 000010  G          #5-64
```

```
                                                                  C  8
VSYM       CREATED BY  MACRO  ON 29-JUN-85 AT 03:07       PAGE 3
```

```
                                                .SBTTL  IMGRES - REWRITE IMAGE
                                        ;+
                                        ;       IMGRES  - REWRITE IMAGES TO SYSTEM IMAGE FILE
                                        ;
                                        ; INPUTS:
                                        ;       $NTLUB  - IMAGE OF UNLOAD BLOCK
                                        ;       $NTLLB  - IMAGE OF LIBRARY BLOCK
                                        ;       $LBB    - LIBRARY BLOCK BIAS
                                        ;       $TMPBF  - LINE TABLE IMAGE
                                        ;
                                        ; OUTPUTS:
                                        ;       BUFFERS WRITTEN BACK TO SYSTEM IMAGE
                                        ;-

000240  005767  000000'                 IMGRES: TST     $FLAG1          ; ANY ERRORS ?
000244  100423                                  BMI     30$             ; NO
000246  005767  000014'                         TST     $SADDR          ; ANY ALLOCATIONS ?
000252  001420                                  BEQ     30$             ; NO

000254  016777  000054' 000000G                 MOV     $UBIAS,@$ADRX   ; SET UP ADDRESS
000262  001414                                  BEQ     30$             ; NO BLOCK TO REWRITE
000264  005077  000000G                         CLR     @$ADRZ          ; ...
000270                                          SWTCHO  #$NTLUB         ; POINT AT UNLOAD BLOCK IMAGE
000276                                          PUTAD   ,#64.           ; REWRITE UNLOAD BLOCK

000310                                          CALL    BUFCK           ; REWRITE LINE TABLE BUFFER
000314                                  30$:    RETURN
```

```
VTMP       CREATED BY  MACRO  ON 29-JUN-85 AT 03:08     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE              REFERENCES

BUFCK      000560 R          9-308     #10-380
FPFX       000713 R         #7-187      18-619
ERRBUF     000565 R         #7-181      18-622
ERRMSG     000560 R         #7-179      18-637
ERR$10   = ******   GX       11-436
ERR$11   = ******   GX       11-437
ERR$12   = ******   GX       11-438
ERR$9    = ******   GX       8-261
ERR1       001050 R          10-393    #10-401
FATAL    = ******   GX       10-393
FL.DDM   = ******   GX       15-538
FL.DLC   = ******   GX       16-555
FL.ECH   = ******   GX       8-253
FL.LLC   = ******   GX       18-615
FMT2       001154 R          7-194      7-195      7-196      7-197      7-198     #7-202
FM.2     = 000000           #7-194     #7-195     #7-196     #7-197     #7-198
F.NRBD   = ******   GX      *8-239     *9-305     *10-341    *10-355    *10-373
F.RSIZ   = ******   GX      *9-306
F1.BBO   = 000001   G       #6-76       13-490     13-495
F1.ERR   = 100000   G       #6-89       18-618
F1.INT   = 000002   G       #6-77
F1.IN1   = 000004   G       #6-78
F1.IN2   = 000010   G       #6-79
F1.IN3   = 000020   G       #6-80
F1.LFI   = 000040   G       #6-81       10-332
F1.LFL   = 000100   G       #6-82
F1.MPL   = 000200   G       #6-83       10-332
F1.MPT   = 000400   G       #6-84
F1.MXT   = 001000   G       #6-85
F1.STE   = 002000   G       #6-86
F1.UMR   = 004000   G       #6-87
F1.VSE   = 010000   G       #6-88
F2.BYT   = 000001   G       #6-97
F2.END   = 000002   G       #6-98       8-262
F2.FIL   = 000004   G       #6-99
F2.LBR   = 000010   G       #6-100
F2.NEG   = 000020   G       #6-101
F2.SUB   = 000040   G       #6-102
IE.EOF   = ******   GX       11-432
IE.RBG   = ******   GX       11-434
IMGRES     000240 R          8-237     #9-297
SYSFDB   = ******   GX      *8-239     *9-305     *9-306     *10-341    *10-355    *10-373
S$$BAS   = ******            7-194      7-194      7-195      7-195      7-196      7-196      7-197      7-197      7-198
                             7-198
TMPBF      000352 R         #6-169      11-418     11-424     12-451
TMPSZ      000556 R         #6-170     *11-423     12-452
TM.RD      001056 R          8-251     #11-418
WPFX       000676 R         #7-186      18-610
$ABFI    = ******   GX       8-280
$ADDR      000020 RG        #6-117
$ADRX    = ******   GX       9-302
$ADRZ    = ******   GX       9-304
```

```
VTMP       CREATED BY  MACRO  ON 29-JUN-85 AT 03:08     PAGE 2
```

```
VTMPIN - VNP TEMPLATE INITIALIZ MACRO V05.03b Saturday 29-Jun-85 03:08  Page 9-1
Symbol table

A$$CHK= 000000            F1.UMR= ****** GX     M$$FCS= 000000        TMPEXT  000000R   002 $LEFT = ****** GX
A$$CPS= 000000            G$$TPP= 000000        M$$MGE= 000000        T$$KMG= 000000        $MXADD= ****** GX
A$$PRI= 000000            G$$TSS= 000000        M$$NET= 000000        T$$MIN= 000000        $MXSIZ= ****** GX
A$$TRP= 000000            G$$TTK= 000000        M$$OVR= 000000        V$$CTR= 001000        $NALL = ****** GX
C$$CKP= 000000            G$$WRD= 000000        N$$ACC= 000001        X$$DBT= 000000        $NAME = ****** GX
C$$ORE= 000400            I$$RAR= 000000        N$$BUF= 000001        ZF.PSE= ****** GX     $NVECT= ****** GX
C$$RSH= 177564            I$$RDN= 000000        N$$LDV= 000001        Z.DAT = ****** GX     $OFF  = ****** GX
D$$BUG= 177514            K$$CNT= 177546        N$$MCP= 000001        Z.FLG = ****** GX     $PDVA = ****** GX
D$$ISK= 000000            K$$CSR= 177546        N$$MLL= 000001        $ADDR = ****** GX     $PRV.N= ****** GX
D$$L11= 000001            K$$LDC= 000000        N$$MOV= 000010        $AZFS = ****** GX     $PTR  = ****** GX
D$$YNC= 000000            K$$TPS= 000074        N$$NCT= 000001        $BFR  = ****** GX     $RADDR= ****** GX
D$$YNM= 000000            K.CSR   000002        N$$PEM= 000001        $BIN  = ****** GX     $SIZE = ****** GX
E$$XPR= 000000            K.PRI   000000        P$$P45= 000000        $CLIOS= ****** GX     $SLTA = ****** GX
FL.CHA= ****** GX         K.VCT   000001        P$$WRD= 000000        $CLQUE= ****** GX     $SZPTR= ****** GX
FL.DDM= ****** GX         LD$LP = 000000        Q$$OPT= 000010        $DFN  = ****** GX     $TM.IN  000000RG
FL LLC= ****** GX         L$$ASG= 000000        RTSPC   000266R       $ERRWZ  000042R   002 $TWARN= ****** GX
FMT0    000072R       002 L$$DRV= 000000        R$$DER= 000000        $ERR8   000004R   002 $TWE  = ****** GX
FMT3    000073R       002 L$$P11= 000001        R$$K11= 000001        $FLAGS= ****** GX     $XBUF2= ****** GX
FM.0  = 000000            L$$11R= 000000        R$$SND= 000000        $FLAG1= ****** GX     $XLEN2= ****** GX
FM.3  = 000000            L.DDS = ****** GX     R$$11M= 000000        $FNOUT= ****** GX     $$CSR = ****** GX
FNDADD  000320R           L.KRBA= ****** GX     SYSFDB= ****** GX     $GETRV= ****** GX     $$VECT= ****** GX
F$$LVL= 000001            M$$CRB= 000124        S$$WRG= 000000        $ICBLN= ****** GX     .MGMGE= ****** GX
F.RSIZ= ****** GX         M$$CRX= 000000        S$$YSZ= 007600

. ABS.  000004    000   (RW,I,GBL,ABS,OVR)
        000444    001   (RW,I,LCL,REL,CON)
DATA    000120    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10375 Words  ( 41 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:11.15
SY:VTMPIN.V2,[132,134]VTMPIN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VTMPIN
```

```
VTMPIN     CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 1
```

```
002212  001460                         BEQ     50$                     ; IF EQ, NO
002214  012705  000000G                MOV     #$BFR,R5                ; POINT AT BUFFER
002220  012704  000012        10$:     MOV     #10.,R4                 ; PRINT 10. WORDS ON A LINE
002224                        20$:     TYPEN   (R5)+,NO,OW             ; PRINT NEXT NUMBER
002262                                 TYPES   < >,NO                  ; INSERT A SPACE
002312  005303                         DEC     R3                      ; MORE IN BUFFER ?
002314  001417                         BEQ     50$                     ; IF EQ, NO
002316  005304                         DEC     R4                      ; ROOM LEFT ON LINE ?
002320  001341                         BNE     20$                     ; IF NE, YES
002322                                 TYPEB   #NEWLIN,NO,#2           ; ELSE, PRINT NEW LINE
002352  000722                         BR      10$                     ; START NEW LINE
002354                        50$:     TYPEB   #NUL,CR,#1              ; INSERT CARRIAGE RETURN
002404                                 RETURN
                                       .DSABL  LSB

                              ;
                              ; MODIFY BUFFER
                              ;
002406     057     040     000 MODPMT: .ASCIZ  %/ %
                                       .EVEN


002412                        MODIFY:  TYPES   <TYPE NUMBER TO MODIFY, CR TO LEAVE UNCHANGED>,LFCR
002442  016703  000000G                MOV     $RSIZE,R3               ; GET SIZE OF BUFFER IN BYTES
002446  062703  000001                 ADD     #1,R3                   ; ROUND TO NEXT WORD
002452  042703  000001                 BIC     #1,R3                   ; ...
002456  006203                         ASR     R3                      ; CONVERT BYTES TO WORDS
002460                                 TYPEN   R3,LF                   ; PRINT SIZE OF BUFFER
002516                                 TYPES   < WORDS IN BUFFER>,CR
002546  012705  000000G                MOV     #$BFR,R5                ; POINT AT BUFFER
002552                        X10:     TYPEN   (R5)+,LF,OW             ; PRINT NEXT NUMBER
002610  012700  002406'                MOV     #MODPMT,R0              ; POINT TO NEW PROMPT STRING
002614                                 CALL    PROMPT                  ; GET NEW VALUE
002620  103432                         BCS     X50                     ; IF CS, EOF
002622  016701  000000G                MOV     $CLIOS,R1               ; POINT TO I/O STATUS BLOCK
002626  005761  000002                 TST     2(R1)                   ; CARRIAGE RETURN ?
002632  001423                         BEQ     X30                     ; IF EQ, YES
002634                                 CALL    $COTB                   ; CONVERT NUMBER AS OCTAL
002640  005702                         TST     R2                      ; VALID INPUT ?
002642  001003                         BNE     X20                     ; IF NE, NO
002644  010165  177776                 MOV     R1,-2(R5)               ; STORE NEW NUMBER
002650  000414                         BR      X30                     ; SKIP TO NEXT WORD
002652                        X20:     TYPES   <ILLEGAL RESPONSE, VALUE LEFT UNCHANGED>,LFCR
002702  005303                X30:     DEC     R3                      ; ANY MORE IN BUFFER ?
002704  001322                         BNE     X10                     ; IF NE, YES
002706                        X50:     RETURN
                              ;
                              ; ALLOCATE DSR
                              ;
002710                        ALL16:   GETNUM  <NUMBER OF BYTES>,R1,X107
002732                                 CALL    $ALL16
002736  103015                         BCC     X106
002740                                 TYPES   <C-BIT SET>,LFCR
002770  000470                         BR      X107
002772  010146                X106:    MOV     R1,-(SP)
002774  010046                         MOV     R0,-(SP)
002776                                 TYPES   <ADDRESS OF NODE >,LF
003026                                 TYPEN   (SP)+,CR,OW
003064                                 TYPEN   (SP)+,LF,OW
```

```

                                ;+                                                                      ;[MP01]
                                ; $ASUM1 - ASSIGN UNIBUS MAPPING REGISTERS                              ;[MP01]
                                ;                                                                       ;[MP01]
                                ; INPUTS :                                                              ;[MP01]
                                ;       URBLCK - MAPPED TO LOWCR FOR THIS CPU                           ;[MP01]
                                ;       R0     - REAL UMR BLOCK ADDRESS                                 ;[MP01]
                                ;       R1     - UMR BLOCK ADDRESS COPY IN VNP IMAGE                    ;[MP01]
                                ;                                                                       ;[MP01]
                                ; OUTPUTS:                                                              ;[MP01]
                                ;       C = 0 - UMR BLOCKS INITIALIZED                                  ;[MP01]
                                ;       C = 1 - CONTIGUOUS UMR ALLOCATION FAILURE                       ;[MP01]
                                ;       R0,R1 - PRESERVED                                               ;[MP01]
                                ;-                                                                      ;[MP01]
                                                                                                        ;[MP01]
                                        .IF DF  R$$MPL                                                  ;[MP01]

                                $ASUM1::MOV     .UMRPT,R4               ; GET UMRPT ADDRESS             ;[MP01]
                                        SWTCHO  #UMRPTR                 ; SET OUTPUT BUFFER             ;[MP01]
                                        PUTAD   R4,#2                   ; UPDATE $UMRPT FOR THIS CPU    ;[MP01]
                                        TST     (R0)+                   ; SKIP LINK WORD                ;[MP01]
                                        TST     (R1)+                   ; ...                           ;[MP01]
                                        MOV     .UMRHD,R5               ; GET $UMRHD ADDRESS FOR THIS CPU ;[MP01]
                                        ADD     #2,R5                   ; GET LISTHEAD+2                ;[MP01]
                                10$:    MOV     R5,R4                   ; SAVE POINTER TO PREVIOUS      ;[MP01]
                                        BEQ     30$                     ; IF EQ - BRANCH                ;[MP01]
                                        SWTCHI  #NEXT                   ; SET INPUT BUFFER              ;[MP01]
                                        GETAD   R5,#M.LGTH              ; READ PREVIOUS UMR BLOCK       ;[MP01]
                                        MOV     NEXT,(R1)               ; POINT TO NEXT FREE UMR        ;[MP01]
                                        MOV     #<M.UMRN-M.UMRA>,R3 ; CALCULATE NEXT FREE UMR           ;[MP01]
                                        ADD     #NEXT,R3                ; ...                           ;[MP01]
                                        ADD     (R3),(R1)               ; ...                           ;[MP01]
                                        MOV     #UBMPR+<31.*4>,-(SP) ; CALCULATE END OF UMR'S           ;[MP01]
                                        DEC     R5                      ; POINT TO NEXT ASSIGNMENT BLOCK ;[MP01]
                                        DEC     R5                      ; ...                           ;[MP01]
                                        SWTCHI                          ; RESET INPUT BUFFER            ;[MP01]
                                        GETAD   R5,#2                   ; READ NEXT UMR BLOCK ADDRESS   ;[MP01]
                                        MOV     $BFR,R5                 ; ...                           ;[MP01]
                                        BEQ     20$                     ; IF NONE - BRANCH              ;[MP01]
                                        GETAD   R5,#2                   ; GET NEXT FREE UMR ADDRESS     ;[MP01]
                                        MOV     $BFR,(SP)               ; ...                           ;[MP01]
                                20$:    SUB     (R1),(SP)               ; CALCULATE NUMBER OF FREE UMR'S * 4 ;[MP01]
                                        CMP     (SP)+,M.UMRN-M.UMRA(R1) ; ENOUGH AVAILABLE ?            ;[MP01]
                                        BLO     10$                     ; IF NOT - BRANCH               ;[MP01]

                                        SWTCHO                          ; RESET OUTPUT BUFFER           ;[MP01]
                                        MOV     R0,$BFR                 ; LINK NEW BLOCK INTO CHAIN     ;[MP01]
                                        DEC     R4                      ; BACK UP TO LINK WORD          ;[MP01]
                                        DEC     R4                      ; ...                           ;[MP01]
                                        PUTAD   R4,#2                   ; UPDATE $UMRHD LIST            ;[MP01]
                                        MOV     (R1),R4                 ; SAVE FIRST UMR ADDRESS        ;[MP01]


```

```
VUNB        CREATED BY  MACRO  ON 29-JUN-85 AT 03:09     PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE             REFERENCES

ER1        000446 R        7-179    #7-182
ER2        000524 R        7-180    #7-184
FATAL   =  ******   GX     7-179     7-180
F.NRBD  =  ******   GX    *6-133    *6-143    *7-155    *7-158
F.RSIZ  =  ******   GX    *6-135    *7-165
F.URBD  =  ******   GX    *6-132    *6-142    *7-154    *7-159
LINK       000166 R        6-130    #7-151
SYSFDB  =  ******   GX    *6-132    *6-133    *6-135    *6-142    *6-143    *7-154    *7-155    *7-158    *7-159
                          *7-165
$ADRX   =  ******   GX     6-114     6-115     6-126     6-134     7-160     7-171
$ADRZ   =  ******   GX     6-111
$ALN18  =  ******   GX     7-156
$BFR    =  ******   GX     6-142     6-143     7-154     7-155
$EROUT  =  ******   GX     7-179     7-180
$GETAD  =  ******   GX     6-135
$PUTAD  =  ******   GX     7-165     7-173
$RSIZE  =  ******   GX    *7-172
$SAVRG  =  ******   GX     6-110
$SAVVR  =  ******   GX     7-151
$UNBAD  =  ******   GX     6-115     6-125    *6-134     7-160     7-162    *7-174
$UNBLK  =  ******   GX     6-118     6-126     6-132     6-133     6-136     7-158     7-159    *7-164     7-166
$UNBSZ  =  ******   GX     7-153
$UNEND  =  ******   GX     6-123
$VUNB      000000 RG      #6-109
$$      =  177777          7-181     7-181    #7-181    #7-182     7-183     7-183    #7-183    #7-184
$$$     =  000450 R       #7-181     7-182    #7-183     7-184
$$$$    =  000524 R       #7-182    #7-184
```

```
                                        .SBTTL  DEAXPT - DEALLOCATE XPT DATA STRUCTURES
                                ;+
                                ;
                                ; DEAXPT - DEALLOCATE XPT DATA STRUCTURES
                                ;
                                ; INPUTS:
                                ;       R3 - ADDRESS OF XPT'S DDB
                                ;       R4 - ADDRESS OF XPT'S PDV
                                ;
                                ; OUPUTS:
                                ;       ALL REG'S PRESERVED
                                ;       XPT DATA STRUCTURES DEALLOCATED
                                ;
                                ;-

000564                          DEAXPT: CALL    $SAVAL                  ;SAVE REG'S
000570                                  GETRV   R3,#N$XLEN              ; Read XPT DDB into $BFR
000606  016701  000036G                 MOV     N$MHC1+$BFR,R1          ; GET ENTREIS IN MINHOP/COST MATRIX
000612  066701  000044G                 ADD     N$MHC2+$BFR,R1          ; INCLUDE AREA ENTRIES
000616  006301                          ASL     R1                      ; CONVERT TO BYTES
000620  062701  000077                  ADD     #77,R1                  ; ROUND TO NEXT BLOCK BOUNDARY
000624  042701  000077                  BIC     #77,R1                  ; ...
000630  000241                          CLC
000632                                  ASR$    6,R1                    ; CONVERT TO BLOCK COUNT
000646  016700  000040G                 MOV     N$MHC1+2+$BFR,R0        ; GET APR BIAS
000652  001411                          BEQ     10$                     ; IF NONE - BRANCH
000654                                  CALL    $DEA18                  ; DEALLOCATE MINHOP/COST vectors
000660                                  GETRV   R3,#N$XLEN              ; READ XPT DDB

000676  016701  000022G         10$:    MOV     N$ROA1+$BFR,R1          ; COPY NUMBER OF ENTRIES IN NODE ROL
000702  001431                          BEQ     20$                     ; IF NONE - BRANCH

000704  066701  000030G                 ADD     N$ROA2+$BFR,R1          ; INCLUDE AREA ROL ENTRIES
000710  006301                          ASL     R1                      ; CONVERT TO BYTES
000712  062701  000077                  ADD     #77,R1                  ; ROUND UP TO NEXT BLOCK
000716  042701  000077                  BIC     #77,R1                  ; ...
000722  000241                          CLC
000724                                  ASR$    6,R1                    ; CONVERT TO BLOCKS
000740  016700  000024G                 MOV     N$ROA1+2+$BFR,R0        ; COPY BIAS
000744                                  CALL    $DEA18                  ; DEALLOCATE POOL..
000750                                  GETRV   R3,#N$XLEN              ; READ XPT

000766  016700  000100G         20$:    MOV     N$TLC+$BFR,R0           ; COPY NUMBER OF CIRCUITS
000772  001432                          BEQ     30$                     ; IF NONE - BRANCH

000774  012701  000046                  MOV     #T$LLEN,R1              ; STORE CIRCUIT COUNTER LENGTH
001000                                  CALL    $MUL                    ; CALCULATE SIZE
001004  062701  000077                  ADD     #77,R1                  ; ROUND UP TO NEXT BLOCK
001010  042701  000077                  BIC     #77,R1                  ; ...
001014  000241                          CLC
001016                                  ASR$    6,R1                    ; CONVERT TO BLOCKS
001032  016700  000102G                 MOV     N$TLC+2+$BFR,R0         ; COPY BIAS
001036                                  CALL    $DEA18                  ; DEALLOCATE POOL..
001042                                  GETRV   R3,#N$XLEN              ; READ XPT DDB

001060  016702  000072G         30$:    MOV     N$ADJ1+$BFR,R2          ; COPY FIRST ADJACENCY BIAS
001064  001457                          BEQ     50$                     ; IF NONE - BRANCH
```

```
                MOV     ICBVL,$RBLCK    ; SET UP BIAS
                CALL    .DEAIC          ; DEALLOCATE ICB
                MOV     NXICB,R0        ; ANOTHER ICB TO DEALLOCATE ?
                BNE     10$             ; IF YES - BRANCH
                                                                        ; ''
                RETURN                                                  ;[MP01]
                .ENDC                                                   ;[MP01]

```

```
                                        .SBTTL  DEV7 - REPLACE TASK AND DEVICE NAMES
                                ;+
                                ; DEV7 - REPLACE REQUESTING TASK'S NAME AND RSX11M DEVICE NAME
                                ;
                                ; INPUTS:
                                ;       $$TASK=REQUESTING TASK'S NAME
                                ;       R1=DEVICE NAME IN ASCII
                                ;       R3=FORMAT STRING ADDRESS
                                ;       R4=ERROR MESSAGE BUFFER ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       R0,R1,R2=DESTROYED
                                ;       R3,R4=UPDATED
                                ;-

003424  010146                  DEV7:   MOV     R1,-(SP)        ; SAVE DEVICE NAME
003426                                  CALL    REP7            ; REPLACE REQUESTING TASK'S NAME
003432  010500                          MOV     R5,R0           ; COPY ERROR MESSAGE BLOCK ADDRESS
003434  005720                          TST     (R0)+           ; GIVING ERROR TEXT ADDRESS
003436  122720  000072          10$:    CMPB    #':,(R0)+       ; LOOK FOR THE COLON
003442  001375                          BNE     10$             ;       ...
003444  005300                          DEC     R0              ; SKIP THE COLON
003446  116640  000001                  MOVB    1(SP),-(R0)     ; DEPOSIT DEVICE NAME
003452  112640                          MOVB    (SP)+,-(R0)     ;       ...
003454                                  RETURN
```

```
                                                                   C  2
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 8

SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

$PUTRC   = ******   GX      24-885      35-1310
$RADDR   = ******   GX     *24-860      24-868     *24-870     *24-872     *26-969     *26-996     *26-1000    *35-1303    *35-1313
$RERR      003250 RG        8-209       8-210       8-211       8-212       8-213       8-214       8-215       8-216       8-217
                            8-218       8-219       8-220       8-221       8-222       8-223       8-224       8-225       8-226
                            8-227       8-228       8-229       8-230       8-231       8-232       8-233       8-234       8-235
                            8-236       8-237      #30-1136
$RSIZE   = ******   G*     *24-858
$SAVAL   = ******   C       26-947      30-1137
$SAVRG   = ******   GX      26-995
$SAVVR   = ******   GX      24-840
$SLN     = ******   GX      14-415     *19-621      29-1100
$SLTA    = ******   GX      14-418     *19-620      22-744      23-773      25-920
$SLTMA   = ******   GX      24-814      24-838      24-892      29-1086
$SLTNM   = ******   GX      24-815      24-893      29-1091
$TIOUT   = ******   GX      30-1156
$TRIB    = ******   GX      14-434     *19-616      19-623
$VFLAG   = ******   GX      14-394      14-408
$$CSR    = ******   GX     *20-679
$$DEV    = ******   GX     *16-518
$$EXT    = ******   GX     *16-509     *16-512
$$INC    = ******   GX     *16-506     *21-728
$$LINX   = ******   GX     *16-520
$$LRB    = ******   GX       9-269      16-503      20-669      27-1022
$$MTP    = ******   GX     *20-703
$$MUX    = ******   GX     *11-321     *19-622
$$PAR    - ******   GX     *16-516     *16-517     *21-729
$$PRI    = ******   GX     *20-686
$$SNUM   = ******   GX     *20-704
$$TBL    = ******   GX     *16-519
$$VECT   = ******   GX     *20-694
.CALI      000224 R         6-93       #6-150
.CCEX      000132 R         6-82       #6-126
.CLIN      000146 R         6-84       #6-130
.CLOG      000170 R         6-95       #6-138
.CNOD      000162 R         6-91       #6-135
.COBJ      000224 R         6-97       #6-152
.CPRO      000136 R         6-83       #6-127
.DALI      000224 R         6-102      #6-157
.DCEX      000224 R         6-98       #6-153
.DCIR      000224 R         6-107      #6-159
.DLIN      000224 R         6-99       #6-154
.DLOG      000224 R         6-103      #6-158
.DMOD      000176 R         6-109      #6-140
.DNOD      000224 R         6-101      #6-156
.DOBJ      000224 R         6-104      #6-160
.DPRO      000224 R         6-100      #6-155
 NOP       000156 R         6-86        6-87        6-88       #6-133
.RNTI      000230 R         6-89       #6-161
.SALI      000224 R         6-92       #6-149
.SCEX      000076 R         6-79       #6-115
.SCIR      000204 R         6-105      #6-142
.SLIN      000116 R         6-81       #6-121

                                                                   D  2
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 9
```

```
          .TITLE  VSTB - VNP COMMAND PROCESSING STATE TABLES
          .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; VNP - COMMAND LINE PARSE TABLES
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
    614                                  .SBTTL  ENTITY ARGUMENT SUBEXPRESSIONS
    615
    616                                  ;        Keyword         Next      Action     Mask (BIS)      Flag to
    617                                  ;        Type, or        State     Routine      Value          Mask
    618                                  ;        Substate
    619
    620                          ;
    621                          ; ENTITY ARGUMENT SUBEXPRESSIONS
    622                          ;
    623                          ; ACTIVE CIRCUITS
    624                          ;
    625 000000                           STATE$  IDACCI
    626 000000                            TRAN$  &K$ACT
    627 000000                           STATE$  ,&P$PEN
    628 000000                            TRAN$  &K$CIRS,        $EXIT,          ,      QF$ACT,        $QUAL
    629                          ;
    630                          ; ACTIVE LINES
    631                          ;
    632 000000                           STATE$  IDACLI
    633 000000                            TRAN$  &K$ACT
    634 000000                           STATE$  ,&P$PEN
    635 000000                            TRAN$  &K$LINS,        $EXIT,          ,      QF$ACT,        $QUAL
    636                          ;
    637                          ; ACTIVE LOGGING
    638                          ;
    639 000000                           STATE$  IDACLO
    640 000000                            TRAN$  &K$ACT
    641 000000                           STATE$  ,&P$PEN
    642 000000                            TRAN$  &K$LOG,         $EXIT,          ,      QF$ACT,        $QUAL
    643                          ;
    644                          ; ACTIVE PROCESSES
    645                          ;
    646 000000                           STATE$  IDACPR
    647 000000                            TRAN$  &K$ACT
    648 000000                           STATE$  ,&P$PEN
    649 000000                            TRAN$  &K$PROS,        $EXIT,          ,      QF$ACT,        $QUAL
    650                          ;
    651                          ; ALIAS
    652                          ;
    653 000000                           STATE$  IDALI
    654 000000                            TRAN$  &K$ALI
    655 000000                           STATE$  ,&P$AID
    656 000000                            TRAN$  !ALIID,         $EXIT
    657                          ;
    658                          ; ALL ALIASES
    659                          ;
    660 000000                           STATE$  IDALAL
    661 000000                            TRAN$  &K$ALL
    662 000000                           STATE$  ,&P$PEN
    663 000000                            TRAN$  &K$ALIS,        $EXIT,          ,      QF$ALL,        $QUAL
    664                          ;
    665                          ; CIRCUIT
    666                          ;
    667 000000                           STATE$  IDCIR
    668 000000                            TRAN$   &K$CIR
    669 000000                           STATE$  ,&P$CRID
    670 000000                            TRAN$   !CIRID,        $EXIT
```

```
   1298 000000                         TRAN$  !STR,          $EXIT,      $STSNM
   1299                        ;
   1300                        ; SINK NODE (FOR SET AND CLEAR LOGGING)
   1301                        ;
   1302 000000                STATE$  OPSIN
   1303 000000                 TRAN$  &K$KNO,        OPSIN1
   1304 000000                 TRAN$  &K$SNK
   1305 000000                STATE$  ,&P$SIN
   1306 000000                 TRAN$  &K$EXE,        $EXIT
   1307 000000                 TRAN$  &K$NOD
   1308 000000                STATE$  ,&P$SIN
   1309 000000                 TRAN$  !OPSHST,       $EXIT,      $STLSH
   1310 000000                 TRAN$  !NODID,        $EXIT,      $STLSN
   1311
   1312                        ;
   1313                        ; "$HOST" FOR SET/CLEAR/SHOW LOGGING SINK NODE
   1314                        ;
   1315
   1316 000000                STATE$  OPSHST
   1317 000000                 TRAN$  <'$>
   1318 000000                STATE$
   1319 000000                 TRAN$  &K$HOS,        $EXIT,      BL.AUT,   OP$HOS,     $OPTON
   1320
   1321                        ;
   1322                        ; NSP SEGMENT SIZE
   1323                        ;
   1324
   1325 000000                STATE$  OPSEG
   1326 000000                 TRAN$  &K$SEG
   1327 000000                STATE$
   1328 000000                 TRAN$  &K$BUF
   1329 000000                STATE$
   1330 000000                 TRAN$  &K$SIZ
   1331 000000                STATE$
   1332 000000                 TRAN$  $NUMBR,$EXIT,$SEG
   1333
   1334                        ;
   1335                        ; INCOMING TIMER VALUE
   1336                        ;
   1337
   1338 000000                STATE$  OPINC
   1339 000000                 TRAN$  &K$INC
   1340 000000                STATE$
   1341 000000                 TRAN$  &K$TIM
   1342 000000                STATE$
   1343 000000                 TRAN$  $NUMBR,$EXIT,$INC
   1344
   1345
   1346                        ;
   1347                        ; OUTGOING TIMER VALUE
   1348                        ;
   1349
   1350 000000                STATE$  OPOUT
   1351 000000                 TRAN$  &K$OUT
   1352 000000                STATE$
   1353 000000                 TRAN$  &K$TIM
   1354 000000                STATE$
```

```
Symbol table

ACC     004044R   002 FL.CNT= ****** G      K$DIS = 000027      K$SCO = 000120           LO.DTE  000047
ALIID   003630R   002 F$$LVL= 000001        K$DTE = 000030      K$SEC = 000121           LO.EVT  000004
ALIID1  003650R   002 G$$TPP= 000000        K$DTES= 000031      K$SEG = 000122           LO.GRO  000024
A$$CHK= 000000        G$$TSS= 000000        K$DUP = 000032      K$SER = 000124           LO.HOS  000040
A$$CPS= 000000        G$$TTK= 000000        K$ECH = 000033      K$SET = 000125           LO.HTM  000032
A$$PRI= 000000        G$$WRD= 000000        K$ENA = 000034      K$SHO = 000126           LO.INA  000062
A$$TRP= 000000        HELTIM  006634R   002 K$EVE = 000035      K$SHU = 000127           LD.INC  000060
BL.AUT  000010R       IDACCI  003102R   002 K$EXE = 000036      K$SIZ = 000133           LD.LDT  000046
BL.MAN  000000R       IDACLI  003122R   002 K$EXI = 000037      K$SNG = 000130           LD.LID  000030
CIRID   003656R   002 IDACLD  003142R   002 K$FAC = 000040      K$SNK = 000131           LD.LIN  000004
CIR1    003700R   002 IDACPR  003162R   002 K$FIL = 000041      K$SNKS= 000132           LO.LTM  000034
CIR2    003720R   002 IDALAL  003220R   002 K$FIR = 000042      K$STAE= 000134           LD.MAC  000022
CIR3    003740R   002 IDALI   003202R   002 K$FIX = 000043      K$STUS= 000135           LO.MDE  000024
CIR4    003750R   002 IDCIR   003240R   002 K$FUL = 000044      K$SUM = 000136           LD.NAM  000044
CKEVEB= ****** G      IDKNAL  003270R   002 K$GLD = 000045      K$SVR = 000123           LD.NET= 000002
CKEVEC= ****** G      IDKNCI  003310R   002 K$GRP = 000046      K$SYS = 000137           LD.NNM  000022
CKEVEN= ****** G      IDKNLI  003330R   002 K$GRPS= 000047      K$TER = 000140           LO.DFL  000012
CKEVER= ****** G      IDKNLO  003350R   002 K$HAL = 000050      K$TIM = 000141           LO.DNM  000004
CLAL    000330R   002 IDKNND  003370R   002 K$HEL = 000051      K$TDP = 000142           LO.OUT  000056
CLALAL  000320R   002 IDKNOB  003410R   002 K$HOS = 000053      K$TRA = 000143           LD.DWN  000026
CLCI    000352R   002 IDKNPR  003430R   002 K$HTM = 000052      K$TRIB= 000144           LD.PAR  000010
CLE     000074R   002 IDLIN   003450R   002 K$INA = 000054      K$UNF = 000145           LO.PRI  000012
CLLI    000362R   002 IDLDG   003466R   002 K$INC = 000055      K$UNI = 000146           LO.RET  000064
CLLO    000372R   002 IDLDNO  003504R   002 K$INS = 000056      K$USE = 000147           LO.RPA  000030
CLLOOP  000404R   002 IDMOD   003524R   002 K$KND = 000057      K$VDT = 000150           LD.SCR  000016
CLNO    000506R   002 IDNOD   003542R   002 K$LGI = 000066      K$VEC = 000151           LD.SEG  000054
CLNOOP  000520R   002 IDOBJ   003570R   002 K$LIN = 000060      K$VER = 000152           LO.SKA  000014
CLOB    000600R   002 IDPRO   003606R   002 K$LINS= 000061      K$X25 = 000153           LD.SKN  000014
CLPR    000610R   002 IDSYS   003624R   002 K$LDA = 000063      K$X29 = 000154           LO.SNM  000035
CLSY    000620R   002 I$$RAR= 000000        K$LDAR= 000064      K$$CNT= 177546           LD.TPA  000020
CM$CLE= ****** G      I$$RDN= 000000        K$LOC = 000065      K$$CSR= 177546           LO.TRB  000034
CM$SET= ****** G      KEYTAB  000000RG  003 K$LOG = 000067      K$$LDC= 000000           LO.TRI  000030
CM$SHO= ****** G      KWRPAS  006644R   002 K$LDD = 000070      K$$TPS= 000074           LD.UNT  000033
COUTIM  006626R   002 KWTPAS  006656R   002 K$LTM = 000062      LD$LP = 000000           LO.VCT  000014
C$$CKP= 000000        K$ACC = 000000        K$MAX = 000071      LD.ALI= 000030           LO.VER  000042
C$$ORE= 000400        K$ACES= 000001        K$MDD = 000072      LD.CEX= 000024           LR.ACC  000020
C$$RSH= 177564        K$ACT = 000002        K$MON = 000073      LD.CIR= 000035           LR.ACO  000052
DESID   003774R   002 K$ADD = 000003        K$MUL = 000074      LD.LIN= 000025           LR.ADD  000012
DESIR1  004060R   002 K$ALI = 000004        K$NAM = 000075      LD.LDG= 000031           LR.ALI  000002
DNODID  004424R   002 K$ALIS= 000005        K$NET = 000076      LD.MOD= 000037           LR.BLK  000010
DNODI1  004452R   002 K$ALL = 000006        K$NETS= 000077      LD.NOD= 000027           LR.CTL  000004
DONE    000070R   002 K$BUF = 000007        K$NDD = 000100      LD.OBJ= 000032           LR.DES  000010
DSTR    007004R   002 K$CHA = 000010        K$NODS= 000101      LD.PRO= 000026           LR.HLP  000002
DSTR.E  000062R       K$CIR = 000011        K$OBJ = 000102      LINID   004076R      002 LR.LIN  000002
D$$BUG= 177514        K$CIRS= 000012        K$DBJS= 000103      LIN1    004120R      002 LR.NAM  000012
D$$ISK= 000000        K$CLE = 000013        K$OFF = 000104      LIN2    004134R      002 LR.PAS  000041
D$$L11= 000001        K$CNSL= 000014        K$ON  = 000105      LISTIM  006640R      002 LR.PRO  000002
D$$YNC= 000000        K$CDN = 000015        K$DUT = 000106      LOGID   004154R      002 LR.STS  000000
D$$YNM= 000000        K$CDNS= 000016        K$OWN = 000107      LO.ADD  000022           LR.TRI  000006
EVEID   006670R   002 K$COP = 000017        K$PAR = 000110      LO.ADR  000052           LR.TYP  000002
EVEID1  006720R   002 K$COS = 000020        K$PAS = 000111      LD.CLS  000002           LR.UCB  000016
EVEID2  006740R   002 K$CDU = 000021        K$PRI = 000112      LO.CON  000006           LR.UIC  000020
EVELIS  006746R   002 K$CSR = 000022        K$PRO = 000113      LO.COP  000010           LR.UNI  000005
EVENUM  006776R   002 K$DEA = 000023        K$PROS= 000114      LD.COS  000020           LS.ALI= 000016
EVERAN  006762R   002 K$DEF = 000024        K$PRTD= 000115      LO.CSR  000010           LS.CEX= 000001
E$$XPR= 000000        K$DES = 000025        K$REC = 000116      LO.CTL  000032           LS.CIR= 000033
FIX     005666R   002 K$DESS= 000026        K$RET = 000117      LD.DES  000002           LS.CXO= 010000
```

```
                                                                   C  7
VSTB          CREATED BY  MACRO  ON 29-JUN-85 AT 03:03       PAGE 11
MACRO CROSS REFERENCE                                         CREF    04.00
MACRO NAME         REFERENCES

                   #13-1076    #13-1078    #13-1083    #13-1085    #13-1087    #13-1088    #13-1090    #13-1092    #13-1099    #13-1101
                   #13-1106    #13-1108    #13-1109    #13-1114    #13-1116    #13-1121    #13-1122    #13-1123    #13-1125    #13-1126
                   #13-1131    #13-1133    #13-1134    #13-1139    #13-1141    #13-1142    #13-1147    #13-1149    #13-1151    #13-1156
                   #13-1158    #13-1160    #13-1165    #13-1167    #13-1169    #13-1174    #13-1176    #13-1178    #13-1183    #13-1185
                   #13-1187    #13-1188    #13-1190    #13-1192    #13-1198    #13-1200    #13-1205    #13-1207    #13-1212    #13-1214
                   #13-1215    #13-1217    #13-1218    #13-1219    #13-1224    #13-1226    #13-1231    #13-1233    #13-1238    #13-1240
                   #13-1245    #13-1247    #13-1252    #13-1254    #13-1259    #13-1260    #13-1262    #13-1264    #13-1265    #13-1267
                   #13-1269    #13-1271    #13-1273    #13-1274    #13-1281    #13-1283    #13-1284    #13-1291    #13-1296    #13-1298
                   #13-1303    #13-1304    #13-1306    #13-1307    #13-1309    #13-1310    #13-1317    #13-1319    #13-1326    #13-1328
                   #13-1330    #13-1332    #13-1339    #13-1341    #13-1343    #13-1351    #13-1353    #13-1355    #13-1362    #13-1364
                   #13-1366    #13-1373    #13-1375    #13-1377    #13-1384    #13-1385    #13-1387    #13-1388    #13-1390    #13-1391
                   #13-1395    #13-1400    #13-1402    #13-1407    #13-1409    #13-1414    #13-1415    #13-1417    #13-1419    #13-1424
                   #13-1426    #13-1427    #13-1432    #13-1434    #13-1439    #13-1441    #13-1442    #13-1443    #13-1448    #13-1450
                   #13-1451    #13-1453    #13-1454    #14-1468    #14-1470    #14-1475    #14-1480    #14-1485    #14-1487    #14-1488
                   #14-1493    #14-1495    #14-1496    #14-1503    #14-1505    #14-1507    #14-1509    #14-1510    #14-1512    #14-1514
                   #14-1515    #14-1516    #14-1518    #14-1520    #14-1525    #14-1526    #14-1531    #14-1533    #14-1535    #14-1540
                   #15-1557    #15-1558    #15-1559    #15-1560    #15-1565    #15-1566    #15-1568    #15-1573    #15-1575    #15-1577
                   #15-1580    #15-1581    #15-1582
VNPDF$             #5-61        5-62
```

```
                                                                   D  7
**FILE**ID**VSYM
```

```
                                                                   C  8
VSYM       CREATED BY  MACRO  ON 29-JUN-85 AT 03:07      PAGE 3
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE                REFERENCES

ZF.MTM  = 000400   G          #5-64
ZF.MUX  = 000040   G          #5-64
ZF.PSE  = 002000   G          #5-64
ZF.SLI  = 010000   G          #5-64
ZF.TIM  = 000200   G          #5-64
ZF.X3P  = 000000   G          #5-64
ZS.ASN  = 100000   G          #5-64
ZS.BSY  = 140000   G          #5-64
Z.AVL     000014   G          #5-64
Z.DAT     000016   G          #5-64
Z.DSP     000000   G          #5-64          5-64
Z.FLG     000010   G          #5-64
Z.LEN   = 000016   G          #5-64
Z.LLN     000006   G          #5-64
Z.MAP     000020   G          #5-64
Z.NAM     000004   G          #5-64
Z.PCB     000012   G          #5-64
Z.SCH     000007   G          #5-64
.ADDRB    000001   G          #5-77
.ADDRW    000002   G          #5-78
.APR      000040   G          #5-108
.BASEB  = 140000   G          #5-63
.BLKB     000003   G          #5-79
.BLKW     000004   G          #5-80
.BYTE     000005   G          #5-81
.CORE     000006   G          #5-82
.CSR      000007   G          #5-83
.CTIM     000010   G          #5-84
.CXALL    000062   G          #5-150
.CXCSR    000004   G          #5-141
.CXLBR    000060   G          #5-149
.CXLB1    000030   G          #5-147
.CXLB2    000044   G          #5-148
.CXLEN    000064   G          #5-151
.CXPCB    000006   G          #5-142
.CXSYM    000010   G          #5-143
.CXUNL    000024   G          #5-145
.CXVEC    000026   G          #5-146
.CXVSM    000016   G          #5-144
.DPRB     000041   G          #5-109
.DPRW     000042   G          #5-110
.DVCHA    000011   G          #5-85
.EOF      000012   G          #5-86
.FILE     000013   G          #5-87
.INT      000014   G          #5-88
.LFILE    000017   G          #5-91
.LIBR     000015   G          #5-89
.LINKS    000016   G          #5-90
.LSTHD    000020   G          #5-92
.LTAB     000021   G          #5-93
.MPTAB    000022   G          #5-94
.MXPTB    000044   G          #5-112
```

```
                                                                   D  8
VSYM       CREATED BY  MACRO  ON 29-JUN-85 AT 03:07      PAGE 4
```

```
    311                                             .SBTTL  $BUFCK - CHECK NEXT 128. BYTES
    312                                     ;+
    313                                             $BUFCK  - BUFFER NEXT 128. BYTES TO LINE TABLE IF NECESSARY
    314                                     ;
    315                                     ; INPUTS:
    316                                     ;       $TMPBF  - IMAGE OF LINE TABLE ( IN VRES )
    317                                     ;       $TMPND  - ADDRESS OF END OF $TMPBF
    318                                     ;       R4      - ADDRESS OF NEXT AVAILABLE BYTE IN $TMPBF
    319                                     ;       $TADDR  - ADDRESS OF LINE TABLE DESTINATION IN SCOM
    320                                     ;       $SIZE   - SIZE OF LINE TABLE
    321                                     ;
    322                                     ; OUTPUT:
    323                                     ;       $TADDR  - NEW DESTINATION ADDRESS ( INCREMENTED )
    324                                     ;       $TMPBF  - SHIFTED LEFT
    325                                     ;       R4      - UPDATED POINTER
    326                                     ;       $SIZE   - UPDATED SIZE ( LEFT )
    327                                     ;-
    328
    329                                             .ENABL  LSB
    330
    331 000316  004267  000000G             $BUFCK::JSR     R2,$SAVVR                       ; SAVE VOLATILE REGISTERS
    332 000322  032767  000240  000000'             BIT     #F1.LFI!F1.MPL,$FLAG1           ; ARE WE DOING MULTIPOINT LINKAGE ?
    333 000330  001107                              BNE     100$                            ; YES, SCRAM
    334 000332  020427  000000C                     CMP     R4,#$TMPBF+<<$TMPND-$TMPBF>/2> ; IS BUFFER AT LEAST HALF-FULL ?
    335 000336  002504                              BLT     100$                            ; NO
    336
    337 000340  012700  000000C                     MOV     #<$TMPND-$TMPBF>/2,R0          ; SET TRANSFER SIZE
    338
    339 000344  020427  000000G             10$:    CMP     R4,#$TMPND                      ; IS BUFFER OVER-FILLED ?
    340 000350  003111                              BGT     110$                            ; YES
    341 000352                                      SWTCHO  #$TMPBF                         ; POINT AT BUFFER TO BE WRITTEN
    342 000360  016767  000016' 000000G             MOV     $TADDR,$RADDR                   ; VIRTUAL ADDRESS
    343 000366  010067  000000G                     MOV     R0,$RSIZE                       ; GET SIZE OF TRANSFER
    344 000372  010001                              MOV     R0,R1                           ; GET SIZE OF TRANSFER
    345 000374  005067  000000G                     CLR     $RBLCK                          ; CLEAR THE BIAS
    346 000400  026727  000016' 120000              CMP     $TADDR,#120000                  ; IS IT A PARTITION LINE TABLE?
    347 000406  103406                              BLO     14$                             ; NO IF LOW
    348 000410  016767  000000G 000000G             MOV     $BIAS,$RBLCK                    ; PROCESS BIAS
    349 000416  042767  160000  000000G             BIC     #160000,$RADDR                  ; CLEAR THE APR BITS
    350 000424  010167  000000G             14$:    MOV     R1,$RSIZE                       ; SET TRANSFER SIZE
    351 000430  026727  000000G 001000              CMP     $RSIZE,#512.                    ; BREAK UP TRANSFER?
    352 000436  101403                              BLOS    15$                             ; BR IF NO
    353 000440  012767  001000  000000G             MOV     #512.,$RSIZE                    ; ELSE ONLY TRANSFER 512. BYTES
    354 000446                              15$:    PUTAD                                   ; WRITE $RSIZE BYTES TO VIRTUAL IMAGE
    355 000452  062767  001000  000000C             ADD     #512.,SYSFDB+F.NRBD+2           ; UPDATE BUFFER ADDRESS TO WRITE FROM
    356 000460  062767  001000  000000G             ADD     #512.,$RADDR                    ; UPDATE ADDRESS TO WRITE TO
    357 000466  162701  001000                      SUB     #512.,R1                        ; UPDATE BYTE COUNT
    358 000472  003354                              BGT     14$                             ; BR IF MORE TO TRANSFER
    359 000474  010067  000000G                     MOV     R0,$RSIZE                       ; GET BACK ORIGINAL BYTE COUNT
    360 000500  060067  000016'                     ADD     R0,$TADDR                       ; UPDATE SCOM ADDRESS POINTER
    361 000504  160067  000012'                     SUB     R0,$SIZE                        ;  AND SIZE
    362 000510  012700  000000G                     MOV     #$TMPBF,R0                      ; PREPARE TO SHIFT LEFT
    363 000514  010401                              MOV     R4,R1                           ; ...
    364 000516  162701  000000C                     SUB     #$TMPBF+<<$TMPND-$TMPBF>/2>,R1  ; ...
    365 000522  003406                              BLE     90$                             ; ...
    366 000524  010002                              MOV     R0,R2                           ; ...
    367 000526  066702  000000G                     ADD     $RSIZE,R2                       ; ...
```

```
                                                                      C 10
VTMP       CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 2

SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE             REFERENCES

$BFALL    000324 RG       #6-157
$BFBAS    000336 RG       #6-161
$BFLBN    000326 RG       #6-158
$BFPTR    000332 RG       #6-159
$BFR    = ******   GX      8-239     10-373    *10-389    *10-390    *10-391     10-401     10-401
$BFXFR    000334 RG       #6-160
$BIAS   = ******   GX      10-348
$BIN      000004 RG       #6-107
$BUFCK    000316 RG       #10-331
$CAPR     000316 RG       #6-154
$CLBUF  = ******   GX     *11-418
$CLDEQ  = ******   GX      8-264
$CLIOS  = ******   GX      11-421
$CLSIZ  = ******   GX     *11-419
$CVAL     000314 RG       #6-153
$DIFF     000300 RG       #6-151
$ELINE    001154 RG        7-194     7-195      7-196      7-197      7-198      8-255     #12-450
$EROUT  = ******   GX      10-393
$ERR13    000720 R        #7-194     13-494
$ERR21    000766 R        #7-195     15-540
$ERR79    001014 R        #7-196     16-557
$ERR82    001042 R        #7-197     17-588
$ERR83    001110 R        #7-198     17-589
$EVEN     001312 RG       #14-517
$FILEN    000302 RG       #6-152
$FILEX  = ******   GX      8-274
$FLAGS  = ******   GX      8-253     15-538     16-555     18-611     18-615
$FLAG1    000000 RG       #6-74     *8-238      9-297     10-332     13-490    *13-495     13-498     17-577    *17-579
                          *18-618
$FLAG2    000002 RG       #6-95      8-246     *8-249      8-262     *8-265
$FNR50    000340 RG       #6-162
$GCLEP  = ******   GX      11-420
$ICBAD    000272 RG       #6-144
$ICBSZ    000274 RG       #6-145
$LBADR    000346 RG       #6-163
$LBIAS    000350 RG       #6-164
$LDA      000060 RG       #6-129
$LEFT     000010 RG       #6-109    *13-492
$LFLD     000322 RG       #6-156
$LINK     000026 RG       #6-120    *10-372
$MPLD     000320 RG       #6-155
$MXADD    000022 RG       #6-118
$MXLEN    000032 RG       #6-122
$MXSIZ    000024 RG       #6-119
$NALL     000034 RG       #6-123
$NLIB     000035 RG       #6-124
$NTLLB    000162 RG       #6-135
$NTLUB    000062 RG       #6-134     9-305
$NVECT    000270 RG       #6-143
$OFF      000030 RG       #6-121
$PTR      000006 RG       #6-108     13-500    *13-501
$PUTAD  = ******   GX      9-306     10-354
```

```
                                                                      D 10
VTMP       CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 3
```

```
                                                                  C 11
VTMPIN     CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE              REFERENCES

FL.CHA  = ******  GX        8-140
FL.DDM  = ******  GX        8-135
FL.LLC  = ******  GX        8-186
FMT0      000072 R          7-87       #7-96
FMT3      000073 R          7-86       #7-97
FM.0    = 000000           #7-87
FM.3    = 000000           #7-86
FNDADD    000320 R          8-170      #9-207
F.RSIZ  = ******  GX       *8-147      *9-208      *9-212
F1.UMP  = ******  GX        8-126
K.CSR     000002            8-147       8-154
K.VCT     000001            8-165
L.DDS   = ******  GX        8-138
L.KRBA  = ******  GX        8-141
RTSPC     000266 R          7-87       #8-181
R$$MPL  = ******            5-60        8-143       8-151       8-161
SYSFDB  = ******  GX       *8-147      *9-208      *9-212
S$$BAS  = ******            7-86        7-86        7-87        7-87
TMPEXT    000000 R         #6-78        8-176
ZF.PSE  = ******  GX        8-133
Z.DAT   = ******  GX       *8-189
Z.FLG   = ******  GX        8-133
$ADDR   = ******  GX       *8-121      *8-138      *9-210      *9-224
$AZFS   = ******  GX        8-177
$BFR    = ******  GX        8-154       8-165       9-209       9-213
$BIN    = ******  GX       *8-118
$CLIOS  = ******  GX        8-179
$CLQUE  = ******  GX        8-178
$DFN    = ******  GX        8-131
$ERRWZ    000042 R         #7-87        8-159       8-173
$ERR8     000004 R         #7-86        8-190
$FLAGS  = ******  GX        8-135      *8-140       8-186
$FLAG1  = ******  GX       *8-126
$FNOUT  = ******  GX        7-86
$GETRV  = ******  GX        8-147       9-208       9-212
$ICBLN  = ******  GX        9-211       9-217
$LEFT   = ******  GX       *8-120
$MXADD  = ******  GX       *8-122
$MXSIZ  = ******  GX       *8-123
$NALL   = ******  GX       *8-128
$NAME   = ******  GX        8-130
$NVECT  = ******  GX       *8-127
$OFF    = ******  GX       *8-124
$PDVA   = ******  GX        8-132       8-188
$PRV.N  = ******  GX        7-86
$PTR    = ******  GX       *8-119
$RADDR  = ******  GX       *8-147      *9-208      *9-212
$SIZE   = ******  GX       *8-125
  [illegible]TA  = ******  GX        8-137
$[illegible]TR  = ******  GX        8-129      *8-129
$[illegible]IN    000000 RG        #8-117
$TWARN  = ******  GX        7-87
```

```
                                                                  D 11
VTMPIN     CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 2
```

```
003122                                    TYPES   < BYTES>,CR
003152                           X107:    RETURN

                                 ;
                                 ; DEALLOCATE DSR
                                 ;
003154                           DEA16:   GETNUM  <ADDRESS>,R5,X108
003200                                    GETNUM  <NUMBER OF BYTES>,R1,X108
003222  010500                            MOV     R5,R0
003224                                    CALL    $DEA16
003230  103014                            BCC     X108
003232                                    TYPES   <C-BIT SET>,LFCR
003262                           X108:    RETURN
                                 ;
                                 ; ALLOCATE 22 BIT ADDRESS
                                 ;
003264                           ALL22:   GETNUM  <NUMBER OF BLOCKS>,R1,X110
003306                                    CALL    $ALL22
003312  103015                            BCC     X109
003314                                    TYPES   <C-BIT SET>,LFCR
003344  000470                            BR      X110
003346  010146                   X109:    MOV     R1,-(SP)
003350  010046                            MOV     R0,-(SP)
003352                                    TYPES   <BIAS OF BUFFER >,LF
003402                                    TYPEN   (SP)+,CR,OW
003440                                    TYPEN   (SP)+,LF,DW
003476                                    TYPES   < BLOCKS>,CR
003526                           X110:    RETURN
                                 ;
                                 ; DEALLOCATE 22 BIT ADDRESS
                                 ;
003530                           DEA22:   GETNUM  <BIAS OF BLOCK>,R5,X111
003554                                    GETNUM  <NUMBER OF BLOCKS>,R1,X111
003576  010500                            MOV     R5,R0
003600                                    CALL    $DEA22
003604  103014                            BCC     X111
003606                                    TYPES   <C-BIT SET>,LFCR
003636                           X111:    RETURN


003640  052767  000000G 000000G  BUFOU:   BIS     #VF.VDT,$VFLAG  ; SET SO WON'T WRITE BUFFERS NEXT TIME
003646  005767  000000C                   TST     SYSFDB+F.FNB+N.FID      ; IS THE IMAGE FILE OPEN?
003652  001464                            BEQ     50$             ; NO IF EQ
003654  032767  000000G 000000G           BIT     #VF.INI,$VFLAG  ; INITIALIZE A SUCCESS?
003662  001460                            BEQ     50$             ; NO IF EQ
                                 ;
                                 ; OUTPUT VALUES STORED IN ROOT SECTION
                                 ;
003664                                    SWTCHO  #$FMASK
003672                                    PUTRC   .FMASK,#2
003712                                    SWTCHO  #$DEVHD
003720                                    PUTRC   .DEVHD
003732                                    SWTCHO  #$PARHD
003740                                    PUTRC   .PARHD
003752                                    SWTCHO  #$CRAVL
003760                                    PUTRC   .CRAVL
003772                                    SWTCHO
004000  032767  000000G 000000G           BIT     #FE.CEX,$FMASK  ; COMM EXEC LOADED?
```

```


        MOV     R5,-2(R1)          ; POINT NEW BLOCK TO NEXT IN LIST          ;[MP01]
        MOV     R4,R5              ; COPY UMR ADDRESS                         ;[MP01]
        BICB    #UBMPR!34,R5       ; MASK OUT COMPLIMENT OF HIGH TO BITS      ;[MP01]
        BIC     R5,R4              ; CLEAR HIGH BYTE AND HI TWO BITS          ;[MP01]
        ASR     R5                 ; SHIFT HIGH TWO BITS TO BITS 4 AND 5      ;[MP01]
        ASLB    R4                 ; GET HIGH BYTE OF LOW 16 BITS             ;[MP01]
        ASL     R4                 ; ...                                      ;[MP01]
        ASL     R4                 ; ...                                      ;[MP01]
        SWAB    R4                 ; FORM LOW 16 BITS                         ;[MP01]
        MOV     R4,M.UMVL-2(R1)    ; SET UP LOW 16 BITS                       ;[MP01]
        MOVB    R5,M.UMVH-2(R1)    ; SET BITS 4 AND 5                         ;[MP01]
30$:    DEC     R1                 ; POINT TO LINK WORD                       ;[MP01]
        DEC     R1                 ; PRESERVE C-BIT                           ;[MP01]
        DEC     R0                 ; SAME                                     ;[MP01]
        DEC     R0                 ; ...                                      ;[MP01]

        RETURN                                                                ;[MP01]
        .ENDC                                                                 ;[MP01]


```

C 14

```
VUNB        CREATED BY  MACRO  ON 29-JUN-85 AT 03:09     PAGE 2

MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              6-130      6-135      7-156      7-165      7-173      7-179      7-180
ERBLK$           #6-105      7-182      7-184
ERMSG$           #6-105      7-181      7-183
ERRPT$           #6-105      7-179      7-180
GETAD            #6-105      6-135
PUTAD            #6-105      7-165      7-173
RAND             #6-135      6-135     #7-165      7-165     #7-173      7-173
RETURN            6-145      7-175
SWTCHI           #6-105      6-132      6-142      7-154      7-159
SWTCHO           #6-105      6-133      6-143      7-155      7-158
```

D 14

```
**FILE**ID**VUNL
```

```
   483 001066  005001                        CLR     R1                      ; INIT REG
   484 001070  156401  000000G               BISB    Z.DAT(R4),R1            ; GET XPT CHANNELS
   485 001074                                GETRV   $DECPT,#D$END           ; READ DECNET HOME BLOCK
   486 001114  066701  000054G               ADD     D$NBRA+$BFR,R1          ; INCLUDE BROADCAST ROUTERS
   487 001120  066701  000056G               ADD     D$NBEA+$BFR,R1          ; INCLUDE BROADCAST ENDNODES
   488 001124  012700  000004                MOV     #A$LEN,R0               ; GET LENGTH OF ENTRY
   489 001130                                CALL    $MUL                    ; CALCULATE TOTAL SIZE
   490 001134  062701  000077                ADD     #77,R1                  ; ROUND UP
   491 001140  042701  000077                BIC     #77,R1                  ; ...
   492 001144  000241                        CLC
   493 001146                                ASR$    6,R1                    ; CONVERT TO BLOCKS
   494 001162  010105                        MOV     R1,R5                   ; SAVE SIZE
   495 001164  010200                        MOV     R2,R0                   ; GET BIAS
   496 001166                                CALL    $DEA18                  ; DEALLOCATE BLOCK
   497 001172                                GETRV   R3,#N$XLEN              ; READ XPT DDB
   498 001210  010501                        MOV     R5,R1                   ; RESTORE SIZE
   499
   500 001212  016700  000074G       40$:    MOV     N$ADJ2+$BFR,R0          ; GET BIAS OF SECOND ADJACENCY D.B.
   501 001216  001402                        BEQ     50$                     ; IF NONE - BRANCH
   502 001220                                CALL    $DEA18
   503
   504 001224                        50$:    RETURN
   505
```

```
 948                                                .SBTTL  VECT - RESET INTERRUPT VECTORS
 949                                        ;
 950                                        ; RESET INTERRUPT VECTOR(S)
 951                                        ;
 952
 953 002402                                 VECT:
 954                                                .IF DF  R$$MPL                                                  ;[MP01]
 955
 956                                                MOV     #KRBMP,R4               ; GET KRB ADDRESS
 957                                                SUB     #K.PRM,R4               ; SET TO OFFSET 0
 958                                                .IFF                                                            ;[MP01]
 959
 960 002402  016504  000000G                        MOV     L.KRBA(R5),R4           ; GET KRB ADDRESS
 961 002406  001550                                 BEQ     50$                     ; IF NO VECTORS - BRANCH
 962 002410                                         GETRV   R4,#$KRBLN
 963                                                .ENDC                                                           ;[MP01]
 964
 965 002426  005000                                 CLR     R0                      ; INIT REG
 966                                                .IF DF  R$$MPL                                                  ;[MP01]
 967                                                CLR     $RBLCK                  ; ASSUME NON MULTIPROCESSOR     ;[MP01]
 968                                                TST     .MPRO                   ; MULTIPROCESSOR ?              ;[MP01]
 969                                                BLT     1$                      ; IF NO - BRANCH                ;[MP01]
 970
 971                                                MOV     K.PRM(R4),R2            ; GET ICB ADDRESS/2 ! CPU NUMBER ;[MP01]
 972                                                BIC     #7777,R2                ; FORM CPU NUMBER               ;[MP01]
 973                                                SWAB    R2                      ; ...                           ;[MP01]
 974                                                ASR     R2                      ; ...                           ;[MP01]
 975                                                ASR     R2                      ; ...                           ;[MP01]
 976                                                ASR     R2                      ; ...                           ;[MP01]
 977                                                ADD     .K6TAB,R2               ; GET ADDRESS OF LOWCR CPU BIAS ;[MP01]
 978
 979                                                SWTCHI                          ; RESET INPUT BUFFER ADDRESS
 980                                                GETRV   R2,#2                   ; READ BIAS                     ;[MP01]
 981                                                MOV     $BFR,$RBLCK             ; COPY BIAS                     ;[MP01]
 982
 983                                        1$:     BISB    K.VCT(R4),R0            ; GET VECTOR ADDRESS            ;[MP01]
 984
 985                                                .IFF                                                            ;[MP01]
 986 002430  156700  000001G                        BISB    K.VCT+$BFR,R0           ; GET VECTOR ADDRESS            ;[MP01]
 987
 988                                                .ENDC                                                           ;[MP01]
 989
 990 002434  006300                                 ASL     R0                      ; CONVERT TO REAL ADDRESS
 991 002436  006300                                 ASL     R0                      ; ...
 992 002440  062700  000000G                        ADD     #$BFR,R0                ; POINT AT DEFAULT BUFFER
 993 002444  010001                                 MOV     R0,R1                   ; GET INTERRUPT VECTOR ADDRESS LIMITS
 994 002446  066701  000000G                        ADD     $MXVCT,R1               ; ...
 995 002452  016703  000010'                        MOV     UBA,R3                  ; GET UNLOAD BLOCK ADDRESS
 996 002456  001525                                 BEQ     101$                    ; IF ZERO, LINE TABLE NOT FOUND!
 997 002460  005723                                 TST     (R3)+                   ; SKIP PDV INDEX
 998 002462  012302                                 MOV     (R3)+,R2                ; GET LINE TABLE STARTING ADDRESS
 999 002464  011303                                 MOV     (R3),R3                 ; AND LENGTH
1000 002466  042703  000003                         BIC     #3,R3                   ; DISCARD STATUS BITS
1001 002472  060203                                 ADD     R2,R3                   ; GIVING END OF LINE TABLE ADDRESS
1002 002474  005067  000000G                        CLR     $RADDR                  ; READ IN FROM VIRTUAL ZERO
1003 002500  026727  000000G 001000                 CMP     $MXVCT,#1000            ; IS VECTOR SPACE WITHIN LEGAL LIMITS ?
1004 002506  003113                                 BGT     121$                    ; IF GT, NO
```

```
                                     .SBTTL  REP7B - REPLACE TASK NAME AND LINE ID
                           ;+
                           ; REP7B - REPLACE REQUESTING TASK'S NAME AND LINE ID
                           ;
                           ; INPUTS:
                           ;         $$TASK=REQUESTING TASK'S NAME
                           ;         R2=SLT ADDRESS
                           ;         R3=FORMAT STRING ADDRESS
                           ;         R4=ERROR MESSAGE BUFFER ADDRESS
                           ;
                           ; OUTPUTS:
                           ;         R0,R1,R2=DESTROYED
                           ;         R3,R4=UPDATED
                           ;-
003456  010246             REP7B:    MOV     R2,-(SP)          ; SAVE SLT ADDRESS
003460                               CALL    REP7              ; REPLACE REQUESTING TASK'S NAME
003464  112314             10$:      MOVB    (R3)+,(R4)        ; LOOK FOR THE SECOND ASTERISK
003466  122724   000052              CMPB    #'*,(R4)+         ;       ...
003472  001374                       BNE     10$               ;       ...
003474  005304                       DEC     R4                ; SKIP THE ASTERISK
003476  010400                       MOV     R4,R0             ; COPY BUFFER ADDRESS
003500  011601                       MOV     (SP),R1           ; GET SLT ADDRESS
003502  116101   000000G             MOVB    L.DDM(R1),R1      ; AND DDM PDV INDEX
003506  066701   000000C             ADD     $PDVTA+AB,R1      ; ADD PDV TABLE ADDRESS
003512  016101   000000G             MOV     AB(R1),R1         ; GET PDV ADDRESS
003516  016146   000000G             MOV     Z.FLG(R1),-(SP)   ; SAVE THE FLAGS WORD
003522  016101   000000G             MOV     Z.NAM(R1),R1      ; GET DDM PROCESS NAME
003526                               CALL    $C5TA             ; CONVERT TO ASCII
003532  122740   000040    20$:      CMPB    #' ,-(R0)         ; DISCARD TRAILING BLANKS
003536  001775                       BEQ     20$               ;       ...
003540  005200                       INC     R0                ;       ...
003542  112720   000055              MOVB    #'-,(R0)+         ; DEPOSIT AN UNDERSCORE
003546  112720   000060              MOVB    #'0,(R0)+         ; AND AN ASCII ZERO
003552  016601   000002              MOV     2(SP),R1          ; GET SLT ADDRESS
003556  116101   000000G             MOVB    L.CTL(R1),R1      ; AND CONTROLLER NUMBER
003562  001404                       BEQ     30$               ; IF ZERO, NO WORK TO DO
003564  005300                       DEC     R0                ; SKIP THE ASCII ZERO
003566  005002                       CLR     R2                ; SUPPRESS LEADING ZEROES
003570                               CALL    $CBOMG            ; CONVERT TO OCTAL
003574  032726   000000G   30$:      BIT     #ZF.MUX,(SP)+     ; IS THIS LINE A MULTIPLEXER ?
003600  001414                       BEQ     40$               ; IF EQ, NO
003602  112720   000055              MOVB    #'-,(R0)+         ; DEPOSIT AN UNDERSCORE
003606  112720   000060              MOVB    #'0,(R0)+         ; AND AN ASCII ZERO
003612  011601                       MOV     (SP),R1           ; ELSE, GET SLT ADDRESS
003614  116101   000000G             MOVB    L.UNT(R1),R1      ; AND UNIT NUMBER
003620  001404                       BEQ     40$               ; IF ZERO, NO WORK TO DO
003622  005300                       DEC     R0                ; BACKUP OVER ASCII ZERO
003624  005002                       CLR     R2                ; SUPPRESS LEADING ZEROES
003626                               CALL    $CBOMG            ; CONVERT TO OCTAL
003632  005726             40$:      TST     (SP)+             ; CLEAN UP THE STACK
003634  010004                       MOV     R0,R4             ; SET NEW BUFFER ADDRESS
003636                               RETURN
```

```
                                                                         D  2
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 9
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE              REFERENCES

.SLOG     000170 R          6-94        #6-137
.SMOD     000216 R          6-108       #6-146
.SNOD     000162 R          6-90        #6-134
.SOBJ     000224 R          6-96        #6-151
.SPRO     000104 R          6-80        #6-117
.TLGTH  = ******  GX        26-996       35-1303     35-1313
.TSKHD  = ******  GX        26-996
..SCSR  = ******  GX       *20-702
```

```
                                                                         E  2
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 10
```

VST
MAC

```
     56                                          .SBTTL  MACROS AND SYMBOLS AND INITIALIZATION
     57                                          .ENABLE LC
     58                                          .DSABLE GBL
     59
     60                                  ; DECLARE EXTERNAL MACROS
     61                                          .MCALL  VNPDF$,ISTAT$,KEYWD$,PRMPT$,STATE$,TRAN$
     62 000000                                   VNPDF$                          ;VNP SYMBOLS
     63
     64         000011                           TAB = 11
     65         000001                           $GPRM   = 1
     66         000001                           $GOFF   = 1     ; ONLY DEFINE OFFSETS FOR PROMPT STRINGS
     67         000001                           $RONLY  = 1     ; MAKE TPARS TABLES READ ONLY
     68
     69                                  ; DECLARE GLOBALS:
     70                                          ; Prompt strings
     71                                          .GLOBL  P$ACT,  P$AID,  P$CLE,  P$CLCI, P$CLLI, P$CLLO
     72                                          .GLOBL  P$CLNO, P$CLOB, P$COP,  P$COS,  P$CRID, P$CSR
     73                                          .GLOBL  P$DEA,  P$DES,  P$DST,  P$DTE,  P$DUP,  P$EVE,  P$GRP                   ;[TD001]
     74                                          .GLOBL  P$HOS,  P$HTM,  P$LGID, P$LIST, P$LNID, P$LOC                           ;[TD001]
     75                                          .GLOBL  P$LOST, P$LTM,  P$MAX,  P$MOID, P$MUL,  P$MXCO                          ;[TD001]
     76                                          .GLOBL  P$MXLI, P$NAD,  P$NAM,  P$NET,  P$NID,  P$NOST, P$OBNA                  ;[TD001]
     77                                          .GLOBL  P$OBUS, P$OBVE, P$OID,  P$OWN,  P$PAR,  P$PEN
     78                                          .GLOBL  P$PID,  P$PRI,  P$SCO,  F$SEAL, P$SECI, P$SELI
     79                                          .GLOBL  P$SELO, P$SENO, P$SEMO, P$SEOB, P$SEPR, P$SET,  P$SHO
     80                                          .GLOBL  P$SIN,  P$SNAM, P$TER,  P$TPAS, P$TRIB, P$VEC
     81                                          .GLOBL  P$VEST, P$VNP,  P$X5ST
     82
     83                                          ; Parse variables
     84                                          .GLOBL  $CMAND, $FLAGS, $OPTON, $QUAL,  $TYPE,  $VFLAG
     85                                          .GLOBL  RQB,    RQBST,  CM$CLE, CM$SET, CM$SHO, .PFLAG
     86                                          .GLOBL  FL.CNT, VF.ADD, VF.ECH, VF.FIX, VF.ON,  VF.PEM
     87                                          .GLOBL  VF.PSI, VF.SED
     88
     89                                          ; Action routines
     90                                          .GLOBL  $ADS,   $ANT,   $CKLOG, $COP,   $COST,  $CSR,   $CTL                    ;[TD001]
     91                                          .GLOBL  $DACC,  $DPAS,  $DUSE,  $EXNOD, $GRO,   $HLPAC
     92                                          .GLOBL  $HTM,   $IFNNA, $IFALI, $IFLOG, $IFXAC, $IFXPR
     93                                          .GLOBL  $IFX5S, $IFX9S, $LNAME, $LTM,   $MACT,  $MDEA
     94                                          .GLOBL  $MXCO,  $MXLI,  $NADD,  $NNAM,  $ONAM,  $OTYP
     95                                          .GLOBL  $OVER,  $OWN,   $PCIR,  $PDT,   $PNAM,  $PRGCL, $PRI
     96                                          .GLOBL  $SCOND, $SCOP1, $SCOP2, $SCOP3, $SCSR,  $STADD, $STLSH
     97                                          .GLOBL  $STALI, $STDES, $STEND, $STHOS, $STLND, $STLSN
     98                                          .GLOBL  $STNAM, $STNAD, $STPAR, $STRPA, $STSNM, $STTPA
     99                                          .GLOBL  $TRI,   $TRIBA, $UNT,   $VEC,   $XMISS, $XTRI
    100                                          .GLOBL  $ARAD1, $ARAD2, $SEG,   $INC,   $OUT,   $INA,   $RET
    101                                          .GLOBL  $XUNT,  CKEVEB, CKEVEC, CKEVEN, CKEVER
    671 000000                          TRAN$   $RAD50,         $EXIT,      $PCIR,      VF.PSI,      $VFLAG
    672                         ;
    673                         ; KNOWN ALIASES
    674                         ;
    675 000000                        STATE$  IDKNAL
    676 000000                         TRAN$  &K$KNO
    677 000000                        STATE$  ,&P$PEN
    678 000000                         TRAN$  &K$ALIS,        $EXIT,          ,      QF$KNO,      $QUAL
    679                         ;
    680                         ; KNOWN CIRCUITS
    681                         ;
    682 000000                        STATE$  IDKNCI
    683 000000                         TRAN$  &K$KNO
    684 000000                        STATE$  ,&P$PEN
    685 000000                         TRAN$  &K$CIRS,        $EXIT,          ,      QF$KNO,      $QUAL
    686                         ;
    687                         ; KNOWN LINES
    688                         ;
    689 000000                        STATE$  IDKNLI
    690 000000                         TRAN$  &K$KNO
    691 000000                        STATE$  ,&P$PEN
    692 000000                         TRAN$  &K$LINS,        $EXIT,          ,      QF$KNO,      $QUAL
    693                         ;
    694                         ; KNOWN LOGGING
    695                         ;
    696 000000                        STATE$  IDKNLO
    697 000000                         TRAN$  &K$KNO
    698 000000                        STATE$  ,&P$PEN
    699 000000                         TRAN$  &K$LOG,         $EXIT,          ,      QF$KNO,      $QUAL
    700                         ;
    701                         ; KNOWN NODES
    702                         ;
    703 000000                        STATE$  IDKNNO
    704 000000                         TRAN$  &K$KNO
    705 000000                        STATE$  ,&P$PEN
    706 000000                         TRAN$  &K$NODS,        $EXIT,          ,      QF$KNO,      $QUAL
    707                         ;
    708                         ; KNOWN OBJECTS
    709                         ;
    710 000000                        STATE$  IDKNOB
    711 000000                         TRAN$  &K$KNO
    712 000000                        STATE$  ,&P$PEN
    713 000000                         TRAN$  &K$OBJS,        $EXIT,          ,      QF$KNO,      $QUAL
    714                         ;
    715                         ; KNOWN PROCESSES
    716                         ;
    717 000000                        STATE$  IDKNPR
    718 000000                         TRAN$  &K$KNO
    719 000000                        STATE$  ,&P$PEN
    720 000000                         TRAN$  &K$PROS,        $EXIT,          ,      QF$KNO,      $QUAL
    721                         ;
    722                         ; LINE
    723                         ;
    724 000000                        STATE$  IDLIN
    725 000000                         TRAN$  &K$LIN
    726 000000                        STATE$  ,&P$LNID
    727 000000                         TRAN$  !LINID,         $EXIT
```

```
   1355 000000                                   TRAN$   $NUMBR,$EXIT,$OUT
   1356
   1357
   1358                               ;
   1359                               ; INACTIVITY TIMER VALUE
   1360                               ;
   1361 000000                                  STATE$   OPINA
   1362 000000                                   TRAN$   &K$INA
   1363 000000                                  STATE$
   1364 000000                                   TRAN$   &K$TIM
   1365 000000                                  STATE$
   1366 000000                                   TRAN$   $NUMBR,$EXIT,$INA
   1367
   1368                               ;
   1369                               ; RETRANSMIT FACTOR
   1370                               ;
   1371
   1372 000000                                  STATE$   OPRET
   1373 000000                                   TRAN$   &K$RET
   1374 000000                                  STATE$
   1375 000000                                   TRAN$   &K$FAC
   1376 000000                                  STATE$
   1377 000000                                   TRAN$   $NUMBR,$EXIT,$RET
   1378
   1379
   1380                               ;
   1381                               ; SINK NODE (FOR SHOW LOGGING)
   1382                               ;
   1383 000000                                  STATE$   OPSINK
   1384 000000                                   TRAN$   &K$KNO,         OPSIN1
   1385 000000                                   TRAN$   &K$SNK
   1386 000000                                  STATE$
   1387 000000                                   TRAN$   &K$EXE,         $EXIT
   1388 000000                                   TRAN$   &K$NOD
   1389 000000                                  STATE$   ,&P$SIN
   1390 000000                                   TRAN$   !OPSHST,        $EXIT,          $STLSH
   1391 000000                                   TRAN$   !NODID,         $EXIT,          $STLSN
   1392
   1393                               ;   (COMMON EXIT FOR KNOWN SINKS)
   1394 000000                                  STATE$   OPSIN1
   1395 000000                                   TRAN$   &K$SNKS,        $EXIT,          ,               OP$KSK,         $OPTON
   1396                               ;
   1397                               ; TRANSMIT PASSWORD
   1398                               ;
   1399 000000                                  STATE$   OPTPAS
   1400 000000                                   TRAN$   !KWTPAS
   1401 000000                                  STATE$   ,&P$TPAS
   1402 000000                                   TRAN$   $STRNG,         $EXIT,          $STTPA
   1403                               ;
   1404                               ; TRIBUTARY
   1405                               ;
   1406 000000                                  STATE$   OPTRI
   1407 000000                                   TRAN$   &K$TRIB
   1408 000000                                  STATE$   ,&P$TRIB
   1409 000000                                   TRAN$   $DNUMB,         $EXIT,          $TRIBA
   1410                               ;
   1411                               ; UNIT CSR
```

```
LS.CX5= 000400          OBJID   004474R     002 OPSEG   006244R     002 OP$OWN= 000020          P$NAM = ****** G
LS.ECH= 001000          OF$INS= 000001          OPSER   006116R     002 OP$PAR= 000010          P$NET = ****** G
LS.FDX= 004000          OF$LOG= 000100          OPSHST  006230R     002 OP$PRI= 000010          P$NID = ****** G
LS.HDX= 010000          OF$OFF= 000002          OPSIN   006160R     002 OP$RET= 010000          P$NOST= ****** G
LS.HLP= 000012          OF$ON = 000000          OPSINK  006330R     002 OP$RPA= 000010          P$OBNA= ****** G
LS.LIN= 000003          OF$SMC= 000200          OPSIN1  006372R     002 OP$SCO= 000001          P$OBUS= ****** G
LS.LMC= 000007          OPA     005572R     002 OPSNAM  006140R     002 OP$SEG= 000400          P$O3VE= ****** G
LS.LOG= 000020          OPADD   005544R     002 OPTPAS  006402R     002 OP$SER= 000400          P$OID = ****** G
LS.MOD= 000036          OPADS   004640R     002 OPTRI   006422R     002 OP$SIN= 000010          P$OWN = ****** G
LS.NOD= 000014          OPAD2   004660R     002 OPUCSR  006440R     002 OP$STA= 000004          P$PAR = ****** G
LS.NTI= 000013          OPANT   004674R     002 OPUCS1  006446R     002 OP$TPA= 000020          P$PEN = ****** G
LS.OBJ= 000022          OPCCSR  004510R     002 OPUSE   006470R     002 OP$TRI= 000040          P$PID = ****** G
LS.OPT= 000400          OPCCS1  004516R     002 OPVEC   006514R     002 OP$UCS= 000020          P$PRI = ****** G
LS.PRO= 000002          OPCIR   004540R     002 OPVER   006574R     002 OP$USE= 000002          P$SCO = ****** G
LS.TOP= 002000          OPCOP   004556R     002 OPVER1  006604R     002 OP$VEC= 000040          P$SEAL= ****** G
LS.TST= 000010          OPCOS   004604R     002 OPX5ST  005360R     002 OP$VER= 000040          P$SECI= ****** G
LS.UNF= 020000          OPCSNM  006134R     002 OP$ADD= 000200          OP$XAC= 000001          P$SELI= ****** G
LS.XIT= 000011          OPDES   004622R     002 OP$ALL= 000200          OP$XPP= 000002          P$SELO= ****** G
LX.ALI= 000017          OPDESV  004624R     002 OP$CCS= 000004          OP$X5S= 000004          P$SEMO= ****** G
LX.CEX= 000004          OPDST   004714R     002 OP$CIE= 004000          OP$X9S= 000010          P$SENO= ****** G
LX.CIR= 000034          OPDST1  004742R     002 OP$CON= 000400          PARTAB  000000RG    002 P$SEOB= ****** G
LX.LIN= 000006          OPDUP   005010R     002 OP$COP= 000001          PAS     004026R     002 P$SEPR= ****** G
LX.LOG= 000021          OPEVE   005040R     002 OP$COS= 000100          PEMSTA  000040R         P$SET = ****** G
LX.NOD= 000015          OPEVE1  005062R     002 OP$DST= 000200          PROID   004502R     002 P$SHO = ****** G
LX.OBJ= 000023          OPGRO   005072R     002 OP$DTE= 000040          P$ACT = ****** G        P$SIN = ****** G
LX.PRO= 000005          OPGRO1  005120R     002 OP$DUP= 000002          P$AID = ****** G        P$SNAM= ****** G
L$$ASG= 000000          OPHA    005176R     002 OP$EST= 000100          P$CLCI= ****** G        P$TER = ****** G
L$$DRV= 000000          OPHOS   005150R     002 OP$EVE= 010000          P$CLE = ****** G        P$TPAS= ****** G
L$$P11= 000001          OPHTM   005130R     002 OP$FIL= 000200          P$CLLI= ****** G        P$TRIB= ****** G
L$$11R= 000000          OPINA   006304R     002 OP$GRP= 000100          P$CLLO= ****** G        P$VEC = ****** G
MODID   004204R     002 OPINC   006260R     002 OP$HOS= 000001          P$CLNO= ****** G        P$VEST= ****** G
MODID5  004220R     002 OPLGST  005334R     002 OP$HTM= 000002          P$CLOB= ****** G        P$VNP = ****** G
MODID9  004262R     002 OPLIN   005210R     002 OP$INA= 004000          P$COP = ****** G        P$X5ST= ****** G
MODI91  004276R     002 OPLNST  005226R     002 OP$INC= 001000          P$COS = ****** G        P$$P45= 000000
M$$CRB= 000124          OPLOC   005272R     002 OP$KDS= 002000          P$CRID= ****** G        P$$WRD= 000000
M$$CRX= 000000          OPLOC1  005312R     002 OP$KDT= 000400          P$CSR = ****** G        QF$ACT= 000001
M$$FCS= 000000          OPLTM   005252R     002 OP$KEV= 001000          P$DEA = ****** G        QF$ALL= 000002
M$$MGE= 000000          OPMACT  005404R     002 OP$KGR= 001000          P$DES = ****** G        QF$KNO= 000004
M$$NET= 000000          OPMDEA  005516R     002 OP$KNT= 010000          P$DST = ****** G        QF$LOO= 000010
M$$OVR= 000000          OPMXCO  005432R     002 OP$KSK= 002000          P$DTE = ****** G        Q$$OPT= 000010
NODADD  004312R     002 OPMXLI  005464R     002 OP$LCT= 000200          P$DUP = ****** G        REJECT  000126R
NODD    004344R     002 OPNAM   005624R     002 OP$LNE= 000040          P$EVE = ****** G        RQB   = ****** G
NODEND  007030R     002 OPNOD   005604R     002 OP$LOC= 000001          P$GRP = ****** G        RQBST = ****** G
NODID   004352R     002 OPNOST  005642R     002 OP$LST= 000002          P$HOS = ****** G        R$$DER= 000000
NODID1  004376R     002 OPOBNA  005712R     002 OP$LTM= 000001          P$HTM = ****** G        R$$K11= 000001
NODID2  004412R     002 OPOBVE  006536R     002 OP$LVL= 000010          P$LGID= ****** G        R$$SND= 000000
NODID3  004420R     002 OPOUT   006272R     002 OP$MAC= 000010          P$LIST= ****** G        R$$11M= 000000
NODNAM  004466R     002 OPOWN   005730R     002 OP$MCO= 000002          P$LNID= ****** G        SEAL    001030R     002
NODX    004324R     002 OPPAR   005746R     002 OP$MDE= 000400          P$LOC = ****** G        SEAL1   001056R     002
N$$ACC= 000001          OPPDT   004752R     002 OP$MLI= 000004          P$LOST= ****** G        SEAL2   001072R     002
N$$BUF= 000001          OPPDT1  005000R     002 OP$MON= 000100          P$LTM = ****** G        SECI    001110R     002
N$$LDV= 000001          OPPRI   005770R     002 OP$MST= 000020          P$MAX = ****** G        SECIOP  001122R     002
N$$MCP= 000001          OPRET   006316R     002 OP$NAM= 000004          P$MOID= ****** G        SELI    001246R     002
N$$MLL= 000001          OPRPAS  006012R     002 OP$NET= 004000          P$MUL = ****** G        SELIOP  001270R     002
N$$MOV= 000010          OPSCO   006032R     002 OP$NLI= 000002          P$MXCO= ****** G        SELO    001402R     002
N$$NCT= 000001          OPSCO1  006042R     002 OP$NOD= 000020          P$MXLI= ****** G        SELOOP  001414R     002
N$$PEM= 000001          OPSCO2  006052R     002 OP$OUT= 002000          P$NAD = ****** G        SEMO    001754R     002
```

```
**FILE**ID**VSYM








VV      VV     SSSSSSSS   YY      YY   MM      MM
VV      VV     SSSSSSSS   YY      YY   MM      MM
VV      VV   SS           YY      YY   MMMM  MMMM
VV      VV   SS           YY      YY   MMMM  MMMM
VV      VV   SS             YY  YY     MM  MM  MM
VV      VV   SS             YY  YY     MM  MM  MM
VV      VV     SSSSSS         YY       MM      MM
VV      VV     SSSSSS         YY       MM      MM
VV      VV           SS       YY       MM      MM
VV      VV           SS       YY       MM      MM
  VV  VV             SS       YY       MM      MM        ....
  VV  VV             SS       YY       MM      MM        ....
    VV       SSSSSSSS         YY       MM      MM        ....
    VV       SSSSSSSS         YY       MM      MM        ....






LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
VSYM - VNP SYMBOL DEFINITIONS   MACRO V05.03b Saturday 29-Jun-85 03:07
```

```
                                                                 D  8
VSYM        CREATED BY  MACRO  ON 29-JUN-85 AT 03:07      PAGE 4
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE              REFERENCES

.MXTAB    000023  G        #5-95
.NODE     000024  G        #5-96
.PECHA    000045  G        #5-113
.PRI      000025  G        #5-97
.PRTCV    000026  G        #5-98
.SBPCB    000066  G        #5-152
.SCOM     000027  G        #5-99
.SECSR    000030  G        #5-100
.SLNB     000031  G        #5-101
.SLNW     000032  G        #5-102
.STNB     000033  G        #5-103
.STNW     000034  G        #5-104
.TIME     000043  G        #5-111
.UNB      000035  G        #5-105
.UNW      000036  G        #5-106
.WORD     000037  G        #5-107
.X2CH1    000061  G        #5-125
.X2CH2    000062  G        #5-126
.X2CH3    000063  G        #5-127
.X2CH4    000064  G        #5-128
.X2CH5    000065  G        #5-129
.X3C01    000046  G        #5-114
.X3C02    000047  G        #5-115
.X3C03    000050  G        #5-116
.X3C04    000051  G        #5-117
.X3C05    000052  G        #5-118
.X3C06    000053  G        #5-119
.X3C07    000054  G        #5-120
.X3C08    000055  G        #5-121
.X3C09    000056  G        #5-122
.X3C10    000057  G        #5-123
.X3C11    000060  G        #5-124
```

```
                                                                 E  8
VSYM        CREATED BY  MACRO  ON 29-JUN-85 AT 03:07      PAGE 5
```

```
000532  112220                    20$:    MOVB    (R2)+,(R0)+             ; SHIFT LEFT
000534  005301                            DEC     R1
000536  001375                            BNE     20$                     ; TILL DONE
000540  010004                    90$:    MOV     R0,R4                   ; RESTORE NEXT FREE BYTE POINTER
000542  066767  000000G 000026'           ADD     $RSIZE,$LINK            ; UPDATE LINK VALUE
000550                            100$:   SWTCHO                          ; POINT BACK AT DEFAULT BUFFER
000556                                    RETURN



                                  ; ALTERNATE ENTRY TO FORCE REMAINDER OF BUFFER TO BE REWRITTEN


000560  004267  000000G           BUFCK:  JSR     R2,$SAVVR               ; SAVE VOLATILE REGISTERS
000564  010400                            MOV     R4,R0                   ; CALCULATE SIZE OF TRANSFER
000566  162700  000000G                   SUB     #$TMPBF,R0              ; ...
000572  000664                            BR      10$                     ; AND GO DO IT


                                  ; ERRORS


000574  012767  000000G 000002G  110$:    MOV     #<$TMPND-$TMPBF>,$BFR+2  ; SIZE OF TEMPLATE PROCESSING BUFFER
000602  010467  000000G                   MOV     R4,$BFR                 ; ADDRESS BUFFER IS FILLED TO
000606  162767  000000G 000000G           SUB     #$TMPBF,$BFR            ; MINIMUM SIZE NEEDED

000614                                    ERRPT$  ERR1,$EROUT,FATAL


                                  ; ERROR MESSAGES


000630                                    ERMSG$  <Insufficient space in template processing buffer%n>
000712                                    ERMSG$  <( TMP$$A PSECT ). Current size is %O bytes, but it%n>
000776                                    ERMSG$  <should be expanded to at least %O bytes.>
001046                                    ERBLK$  ERR1,<$BFR+2,$BFR>

                                          .DSABL  LSB
```

```
                                                                   D 10
VTMP        CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 3
SYMBOL CROSS REFERENCE                                    CRFF    04.00

SYMBOL   VALUE            REFERENCES

$RADDR   = ******  GX     *10-342     *10-349     *10-356
$RBLCK   = ******  GX     *10-345     *10-348
$REP.C   = ******  GX      7-194       7-195       7-196       7-197       7-198
$RQ1       001226 RG      #13-489
$RQ2       001220 RG      #13-487      14-520
$RQ3       001212 RG      #13-485
$RQ4       001204 RG      #13-483
$RQ5       001176 RG      #13-481
$RQ7       001170 RG      #13-479
$RSIZE   = ******  GX     *10-343     *10-350      10-351     *10-353     *10-359      10-367      10-372
$SADDR     000014 RG      #6-115      *8-224       9-299
$SAVAL   = ******  GX      8-245       18-620
$SAVVR   = ******  GX      10-331      10-380
$SIZE      000012 RG      #6-114       8-248      *10-361      14-518     *14-524
$SZPTR     000036 RG      #6-125      *8-247       8-248      *8-266       17-581      17-581
$TADDR     000016 RG      #6-116       10-342      10-346     *10-360
$TERR      001470 RG       7-194       7-195       7-196       7-197       7-198      #18-617
$TIOUT   = ******  GX      12-453      18-640
$TMALT   = ******  GX      8-231
$TMCHK   = ******  GX      8-230
$TMFIL     000066 R        8-228      #8-245       8-273
$TMFIX     000220 RG      #8-272
$TMLTA   = ******  GX      8-236
$TMP       000000 RG      #8-223
$TMPBF   = ******  GX      10-334      10-334      10-337      10-341      10-362      10-364      10-364      10-382      10-389
                           10-391
$TMPKW   = ******  GX      8-257
$TMPND   = ******  GX      10-334      10-337      10-339      10-364      10-389
$TMPST   = ******  GX      8-258
$TMVCT   = ******  GX      8-234
$TM.IN   = ******  GX      8-225
$TPARS   = ******  GX      8-227       8-281
$TWARN     001444 RG      #18-609      18-616
$TWE       001460 RG      #18-614
$UBIAS     000054 RG      #6-127       9-302
$UDA       000056 RG      #6-128
$VALUE     000276 RG      #6-150      *8-250
$VECT1     000262 RG      #6-140
$VECT2     000264 RG      #6-141
$VECT3     000266 RG      #6-142
$VEXP    = ******  GX      8-232
$VFDDM     001344 RG      #15-537
$VFDLC     001364 RG      #16-554
$VFLV1     001416 RG      #17-580
$VFSPC     001404 RG      #17-576
$$       = 177777          10-398      10-398     #10-398      10-399      10-399     #10-399      10-400      10-400     #10-400
                          #10-401
$$$      = 000630 R       #10-398      10-401
$$$$     = 001050 R       #10-401
.ABFI      000234 RG      #8-279
.BYTE    = *****   GX      14-522
.TPARS   = *****   GX      8-259
```

```
                                                                   E 10
VTMP        CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 4
```

```
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE               REFERENCES

$TWE     = ******  GX          7-86
$XBUF2   = ******  GX          8-118        8-119
$XLEN2   = ******  GX          8-120
$$CSR    = ******  GX          8-154
$$VECT   = ******  GX          8-171
.MGMGE   = ******  GX          9-214
```

```
    456 004006  001406                          BEQ     50$                     ; NO IF EQ
    457 004010  016746  000000G                 MOV     $FMASK,-(SP)            ; SAVE THE MASK
    458 004014                                  CALL    $VIMO                   ; OUTPUT COMM EXEC
    459 004020  012667  000000G                 MOV     (SP)+,$FMASK
    460 004024                          50$:    RETURN
    461
    462                                 ;
    463                                 ; USE GCL TO GET INPUT FROM USER
    464                                 ;
    465 004026                          PROMPT:
    466 004026  016746  000000G                 MOV     $CLPMT,-(SP)                    ; SAVE CURRENT PROMPT STRING
    467 004032  010067  000000G                 MOV     R0,$CLPMT                       ; SET NEW PROMPT STRING
    468 004036                                  GCL$                                    ; GET COMMAND LINE
    469 004042  103411                          BCS     20$                             ; IF CS, EOF
    470 004044  016700  000000G                 MOV     $CLBUF,R0                       ; POINT TO BUFFER
    471 004050  122720  000040          10$:    CMPB    #40,(R0)+                       ; SKIP BLANKS
    472 004054  001775                          BEQ     10$                             ; IF EQ, MORE
    473 004056  005300                          DEC     R0                              ; WENT TOO FAR
    474 004060  010067  000000G                 MOV     R0,$CLBUF                       ; UPDATE BUFFER POINTER
    475 004064  000241                          CLC                                     ; MAKE SURE CARRY CLEAR !
    476 004066  012667  000000G         20$:    MOV     (SP)+,$CLPMT                    ; RESTORE PREVIOUS PROMPT
    477 004072                                  RETURN
    478                                 ;
    479                                 ; CVBTO - CONVERT BINARY TO OCTAL
    480                                 ;
    481 004074                          CVBTO:
    482 004074  016700  000000G                 MOV     $CLBUF,R0                       ; POINT TO OUTPUT BUFFER
    483 004100                                  CALL    $CBTA                           ; CONVERT BINARY TO OCTAL
    484 004104  016701  000000G                 MOV     $CLBUF,R1                       ; POINT TO START OF BUFFER
    485 004110  160100                          SUB     R1,R0                           ; CALCULATE LENGTH OF STRING
    486 004112                                  RETURN
    487
    488 004114                                  ENTEXT                                  ; END OF TEXT STRINGS
    489
    490         000001                          .END
```

```
                                 ;+
                                 ; $DEUMR - DE-ASSIGN UNIBUS MAPPING REGISTERS
                                 ;
                                 ; THIS ROUTINE IS CALLED TO SCAN THE LIST OF CURRENTLY ALLOCATED UMR'S
                                 ; ($UMRHD) AND TO UNLINK THE CALLER'S UMR BLOCK.
                                 ;
                                 ; INPUTS:
                                 ;          R2=UMR BLOCK REAL MEMORY ADDRESS
                                 ;          .UMRHD=VALUE OF $UMRHD FROM RSX11M.STB FILE
                                 ;
                                 ;          ON MULTIPROCESSOR RSX-11M-PLUS SYSTEMS, .CPBIA MUST BE SET TO
                                 ;          THE BIAS THE CPU'S DATA AREA.
                                 ;
                                 ; OUTPUTS:
                                 ;          C-BIT=SUCCESS/FAILURE
                                 ;          ALL REGISTERS ARE PRESERVED
                                 ;-
000172                           $DEUMR::CALL    $SAVAL          ; SAVE ALL REGISTERS
000176  010200                           MOV     R2,R0           ; COPY BLOCK ADDRESS
000200  062700  000002                   ADD     #2,R0           ; EACH BLOCK POINTS AT NEXT+2
000204                                   CALL    GFIRST          ; GET FIRST UMR BLOCK(S)
000210  000402                           BR      20$
000212                           10$:    CALL    GNEXT           ; GET NEXT UMR BLOCK
000216  103422                   20$:    BCS     30$             ; IF CS, FAILURE
000220  026700  000000'                  CMP     PREV,R0         ; DOES PREVIOUS BLOCK POINT AT OURS?
000224  001372                           BNE     10$             ; IF NE, NO
000226  162700  000002                   SUB     #2,R0           ; RESTORE ORIGINAL BLOCK ADDRESS
000232  020003                           CMP     R0,R3           ; IS OUR BLOCK THE NEXT ONE?
000234  001022                           BNE     101$            ; IF NE, NO - VERY BAD
000236  016767  000014' 000000'          MOV     NEXT,PREV       ; COPY OUR POINTER INTO HIS
000244                                   SWTCHO  #PREV           ; CHANGE DEFAULT BUFFER

                                         .IF DF  R$$MPL                                          ;[MP01]
                                         TST     .MPRO           ; MULTIPROCESSOR SYSTEM ?       ;[MP01]
                                         BLT     25$             ; IF NO - BRANCH                ;[MP01]
                                         MOV     $CPBIA,$RBLCK   ; MAP TO CPU BIAS               ;[MP01]
                                         PUTAD   R2              ; REWRITE PREVIOUS BLOCK        ;[MP01]
                                         BR      27$             ; CONTINUE                      ;[MP01]
                                         .ENDC                                                   ;[MP01]

000252                           25$:    PUTRC   R2              ; RE-WRITE PREVIOUS BLOCK
000262  000241                   27$:    CLC                     ; INDICATE SUCCESS
000264                           30$:    SWTCHI                  ; RESET DEFAULT BUFFERS
000272                                   SWTCHO                  ;       ...
000300                                   RETURN

                                 ;
                                 ; ERROR CONDITION
                                 ;
000302                           101$:   CRASH                   ; UMR BLOCKS ARE SCREWED UP!
```

```
**FILE**ID**VUNL
```

```
VV      VV   UU      UU   NN      NN   LL
VV      VV   UU      UU   NN      NN   LL
VV      VV   UU      UU   NN      NN   LL
VV      VV   UU      UU   NN      NN   LL
VV      VV   UU      UU   NNNN    NN   LL
VV      VV   UU      UU   NNNN    NN   LL
VV      VV   UU      UU   NN  NN  NN   LL
VV      VV   UU      UU   NN  NN  NN   LL
VV      VV   UU      UU   NN    NNNN   LL
VV      VV   UU      UU   NN    NNNN   LL
  VV  VV     UU      UU   NN      NN   LL               ....
  VV  VV     UU      UU   NN      NN   LL               ....
    VV       UUUUUUUUUU   NN      NN   LLLLLLLLLL       ....
    VV       UUUUUUUUUU   NN      NN   LLLLLLLLLL       ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                              .SBTTL  DEAECL - DEALLOCATE ECL NODE COUNTERS

                                      ;+
                                      ;
                                      ; DEAECL - DEALLOCATE ECL NODE COUNTER BLOCK
                                      ;
                                      ; INPUTS:
                                      ;       R3 - VIRTUAL ADDRESS OF ECL'S DDB
                                      ;       R4 - ADDRESS OF ECL'S PDV
                                      ;
                                      ; OUTPUTS:
                                      ;       R3 - PRESERVED
                                      ;       R4 - PRESERVED
                                      ;       ECL NODE COUNTER BLOCK IS DEALLOCATED
                                      ;
                                      ;-
001226                                DEAECL:
001226                                        GETRV   R3,#N$ELEN              ; Read ECL DDB into $BFR
001244  016701  000044G                       MOV     N$ENC+2+$BFR,R1         ; GET START OF LAST NODE COUNTER BLOCK
001250  001422                                BEQ     10$                     ; IF NO COUNTERS - BRANCH
001252  062701  000050                        ADD     #E$NLEN,R1              ; POINT TO END OF COUNTER BLOCK
001256  166701  000042G                       SUB     N$ENC+$BFR,R1           ; CALCULATE NUMBER OF BYTES
001262  062701  000077                        ADD     #77,R1                  ; ROUND TO NEXT BLOCK BOUNDARY
001266  042701  000077                        BIC     #77,R1                  ; ...
001272                                        ASR$    6,R1                    ; CONVERT TO BLOCK COUNT
001306  016700  000046G                       MOV     N$ENC+4+$BFR,R0         ; GET APR BIAS
001312                                        CALL    $DEA18                  ; DEALLOCATE NODE COUNTCRS
001316                                10$:    RETURN

```

```
1005
1006                                               .IF NDF R$$MPL                                         ;[MP01]
1007 002510                                        GETRV    ,$MXVCT          ; AN IMAGE OF VECTOR SPACE
1008
1009                                               .IFF                                                   ;[MP01]
1010                                               GETAD    ,$MXVCT          ; READ VECTOR AREA           ;[MP01]
1011
1012                                               .ENDC                                                  ;[MP01]
1013
1014 002522 005067 000174'                         CLR      ICBSAV           ; START ADDRESS OF ICB
1015 002526 005067 000176'                         CLR      ICBSZ            ; SIZE OF AN ICB
1016 002532 005067 000200'                         CLR      PRLTA            ; FLAG
1017 002536 020227 120000                          CMP      R2,#120000       ; PARTITION LINE TABLE ALLOCATION
1018 002542 103403                                 BLO      10$              ; NO IF LOW
1019 002544 152767 000001 000200'                  BISB     #PR.LTA,PRLTA    ; SET FLAG SO WILL KNOW LATER
1020 002552                               10$:     CALL     ICBSC            ; SEE IF VECTOR POINTS AT ICB
1021 002556 103445                                 BCS      40$              ; NONE IF SET
1022 002560 152767 000002 000200'                  BISB     #PR.SEC,PRLTA    ; INDICATE BEEN THRU VECSC ONCE OR MORE
1023 002566 132767 000001 000200'                  BITB     #PR.LTA,PRLTA    ; PARTITION LINE TABLE?
1024 002574 001402                                 BEQ      30$              ; NO IF EQ
1025 002576 011067 000174'                         MOV      (R0),ICBSAV      ; START OF ICB
1026
1027 002602                               30$:
1028 002602 010004                                 MOV      R0,R4            ; COPY VECTOR ADDRESS
1029 002604 162704 000000G                         SUB      #$BFR,R4         ; CONVERT TO REAL VECTOR ADDRESS
1030 002610 042704 000077                          BIC      #77,R4           ; DISCARD LOW 6 BITS
1031 002614                                        ASR$     5,R4             ; CONVERT TO A WORD INDEX
1032 002626 017404 000130'                         MOV      @NSITAB(R4),R4   ; GET APPROPRIATE VECTOR CONTENTS
1033 002632 010046                                 MOV      R0,-(SP)         ; COPY VECTOR ADDRESS AGAIN
1034 002634 162716 000000G                         SUB      #$BFR,(SP)       ; AGAIN CONVERT TO REAL VECTOR ADDRESS
1035 002640 042716 177700                          BIC      #^C<77>,(SP)     ; GET LOW 6 BITS
1036 002644 006216                                 ASR      (SP)             ; DIVIDE BY 4
1037 002646 006216                                 ASR      (SP)             ; ...
1038 002650 062716 000000G                         ADD      #PR7,(SP)        ; GIVING DEFAULT PRIORITY
1039 002654 010420                                 MOV      R4,(R0)+         ; RESTORE THE VECTOR
1040 002656 012620                                 MOV      (SP)+,(R0)+      ; ...
1041 002660 020001                                 CMP      R0,R1            ; DONE?
1042 002662 103003                                 BHIS     40$              ; IF HIS, YES
1043 002664                                        CALL     ICBSC            ; DOES NEXT VECTOR POINT AT OUR LINE TABLE?
1044 002670 103344                                 BCC      30$              ; IF CC, YES
1045 002672                               40$:
1046 002672 005067 000000G                         CLR      $RADDR           ; WRITE FROM VIRTUAL ZERO
1047 002676                                        SWTCHO                    ; USE DEFAULT BUFFER TO OUTPUT
1048
1049                                               .IF NDF R$$MPL                                         ;[MP01]
1050 002704                                        PUTRC    #0,$MXVCT        ; REWRITE UPDATED VECTOR IMAGE
1051                                               .IFF                                                   ;[MP01]
1052                                               PUTAD    #0,$MXVCT        ; REWRITE UPDATED VECTOR IMAGE ;[MP01]
1053
1054                                               .ENDC                                                  ;[MP01]
1055
1056 002724 005700                                 TST      R0               ; WAS AT LEAST 1 VECTOR FOUND?
1057 002726 001402                                 BEQ      111$             ; IF EQ, NO
1058 002730                               60$:     RETURN
1059
1060                                      ;
1061                                      ; ERROR CONDITIONS
```

```
   1289                                                .SBTTL  DEQNTI - DEALLOCATE BLOCK QUEUED TO NTINIT
   1290                                         ;+
   1291                                         ; *** - DEQNTI - DEALLOCATE BLOCK QUEUED TO NTINIT, AND REMOVE
   1292                                         ;              THE ENTRY FROM THE CLOCK QUEUE.
   1293                                         ;
   1294                                         ; INPUTS:
   1295                                         ;       R5 = NTINIT'S TCB ADDRESS
   1296                                         ;
   1297                                         ; OUTPUTS:
   1298                                         ;       NONE.
   1299                                         ;
   1300                                         ;-
   1301
   1302 003640                                  DEQNTI:
   1303 003640                                          GETRV   R5,.TLGTH               ; READ IN NTINIT'S TCB
   1304 003656  016700  000012G                         MOV     T.RCVL+$BFR,R0          ; POINT AT THE SEND DATA PACKET
   1305 003662  001413                                  BEQ     10$                     ; IF EQ, ACP NOT MOUNTED
   1306 003664  005067  000012G                         CLR     T.RCVL+$BFR             ; REMOVE ENTRY FROM NTINIT'S QUEUE
   1307 003670  010504                                  MOV     R5,R4                   ;   COPY THE TCB ADDRESS
   1308 003672  062704  000012                          ADD     #T.RCVL,R4              ;   CALCULATE ADDRESS OF QUEUE
   1309 003676  010467  000014G                         MOV     R4,T.RCVL+2+$BFR        ;   POINT QUEUE AT ITSELF
   1310 003702                                          PUTRC                           ; REWRITE THE TCB
   1311 003706                                          CALL    $DEA16                  ; DE-ALLOCATE THE SEND DATA PACKET
   1312 003712                                  10$:    CALL    CLQREM                  ; REMOVE RUN REQUESTS FROM CLOCK QUEUE
   1313 003716                                          GETRV   R5,.TLGTH               ; PUT NTINIT'S TCB BACK IN $BFR
   1314 003734                                          RETURN
   1315
   1316         000001                                  .END
```

```
                                                                E  2
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 10

MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME       REFERENCES

ASL$             #5-59     20-685
CALL              9-282    14-399    14-411    15-470    17-535    18-569    19-592    19-618    22-743    24-850
                  24-863   24-870    24-885    24-903    26-947    26-954    26-957    26-969    26-996    26-1003
                  30-1137  30-1143   30-1156   30-1159   32-1193   32-1206   33-1226   34-1251   34-1263   34-1274
                  34-1284  35-1303   35-1310   35-1311   35-1312   35-1313
CALLR            #5-60     24-809
DCBDF$           #5-60     5-64
EMSG$            #5-59     24-850    24-903
EMSG$R           #5-59     9-294     11-323    12-360    13-379    14-447    14-448    14-449    14-450    14-451
                  15-482   15-483    19-640    19-641    19-642    19-643    20-711    20-712    20-713    20-714
                  22-751   23-776    24-805    24-820    24-877    25-934    26-975    29-1123
GETAD            #5-60
GETRV            #5-59     24-863    24-870    26-969    26-996    26-1003   35-1303   35-1313
NTLER$           #5-59     8-209     8-210     8-211     8-212     8-213     8-214     8-215     8-216     8-217
                  8-218    8-219     8-220     8-221     8-222     8-223     8-224     8-225     8-226     8-227
                  8-228    8-229     8-230     8-231     8-232     8-233     8-234     8-235     8-236     8-237
PUTRC            #5-59     24-885    35-1310
RAND             #24-863   24-863   #24-870    24-870   #24-885    24-885   #26-969    26-969   #26-996    26-996
                 #26-1003  26-1003  #35-1303   35-1303  #35-1310   35-1310  #35-1313   35-1313
RESRG            #5-60
RETURN            9-289    10-302    11-325    11-340    12-356    13-375    14-443    15-477    16-521    17-548
                  18-571   19-635    20-706    21-730    22-759    23-777    24-847    24-907    25-929    26-978
                  26-1008  27-1032   28-1064   29-1117   30-1161   31-1176   32-1208   33-1234   34-1287   35-1314
SAVRG            #5-60
SWTCHI           #5-59     9-287     24-869
SWTCHO           #5-60     9-288     24-884
TCBDF$           #5-60     5-66
UCBDF$           #5-60     5-65
VNPDF$           #5-59     5-63
```

```
                                                                F  2
**FILE**ID**VSET
```

```
   103                                     .SBTTL  KEYWORDS
   104                             ;****
   105                             ; TPARS STATE TABLES
   106                             ;****
   107
   108 000000                              ISTAT$  PARTAB,KEYTAB
   109                             ;
   110                             ; DEFINE KEYWORDS
   111                             ;
   112 000000                              KEYWD$  "ACCOUNT",              K$ACC
   113 000000                              KEYWD$  "ACCESS",               K$ACES
   114 000000                              KEYWD$  "ACTIVE",               K$ACT
   115 000000                              KEYWD$  "ADDRESS",              K$ADD
   116 000000                              KEYWD$  "ALIAS",                K$ALI
   117 000000                              KEYWD$  "ALIASES",              K$ALIS
   118 000000                              KEYWD$  "ALL",                  K$ALL
   119 000000                              KEYWD$  "BUFFER",               K$BUF
   120 000000                              KEYWD$  "CHARACTERISTICS",      K$CHA
   121 000000                              KEYWD$  "CIRCUIT",              K$CIR
   122 000000                              KEYWD$  "CIRCUITS",             K$CIRS
   123 000000                              KEYWD$  "CLEAR",                K$CLE
   124 000000                              KEYWD$  "CONSOLE",              K$CNSL
   125 000000                              KEYWD$  "CONTROLLER",           K$CON
   126 000000                              KEYWD$  "CONTROLLERS",          K$CONS
   127 000000                              KEYWD$  "COPIES",               K$COP
   128 000000                              KEYWD$  "COST",                 K$COS
   129 000000                              KEYWD$  "COUNTER",              K$COU
   130 000000                              KEYWD$  "CSR",                  K$CSR
   131 000000                              KEYWD$  "DEAD",                 K$DEA
   132 000000                              KEYWD$  "DEFAULT",              K$DEF
   133 000000                              KEYWD$  "DESTINATION",          K$DES
   134 000000                              KEYWD$  "DESTINATIONS",         K$DESS
   135 000000                              KEYWD$  "DISABLE",              K$DIS
   136 000000                              KEYWD$  "DTE",                  K$DTE
   137 000000                              KEYWD$  "DTES",                 K$DTES
   138 000000                              KEYWD$  "DUPLEX",               K$DUP
   139 000000                              KEYWD$  "ECHO",                 K$ECH
   140 000000                              KEYWD$  "ENABLE",               K$ENA
   141 000000                              KEYWD$  "EVENTS",               K$EVE
   142 000000                              KEYWD$  "EXECUTOR",             K$EXE
   143 000000                              KEYWD$  "EXIT",                 K$EXI
   144 000000                              KEYWD$  "FACTOR",               K$FAC
   145 000000                              KEYWD$  "FILE",                 K$FIL
   146 000000                              KEYWD$  "FIRSTFIT",             K$FIR
   147 000000                              KEYWD$  "FIXED",                K$FIX
   148 000000                              KEYWD$  "FULL",                 K$FUL
   149 000000                              KEYWD$  "GLOBAL",               K$GLO
   150 000000                              KEYWD$  "GROUP",                K$GRP
   151 000000                              KEYWD$  "GROUPS",               K$GRPS
   152 000000                              KEYWD$  "HALF",                 K$HAL
   153 000000                              KEYWD$  "HELP",                 K$HEL
   154 000000                              KEYWD$  "HELLO",                K$HTM
   155 000000                              KEYWD$  "HOST",                 K$HOS
   156 000000                              KEYWD$  "INACTIVITY",           K$INA
   157 000000                              KEYWD$  "INCOMING",             K$INC
   158 000000                              KEYWD$  "INSPECT",              K$INS
   159 000000                              KEYWD$  "KNOWN",                K$KNO
```

ENTITY ARGUMENT SUBEXPRESSIONS

```
    728                                        ;
    729                                        ; LOGGING
    730                                        ;
    731 000000                                         STATE$  IDLOG
    732 000000                                          TRAN$  &K$LOG
    733 000000                                         STATE$  ,&P$LGID
    734 000000                                          TRAN$  !LOGID,          $EXIT
    735                                        ;
    736                                        ; LOOP NODES
    737                                        ;
    738 000000                                         STATE$  IDLONO
    739 000000                                          TRAN$  &K$LOO
    740 000000                                         STATE$  ,&P$PEN
    741 000000                                          TRAN$  &K$NODS,         $EXIT,          ,       QF$LOO,         $QUAL
    742                                        ;
    743                                        ; MODULE
    744                                        ;
    745 000000                                         STATE$  IDMOD
    746 000000                                          TRAN$   &K$MOD
    747 000000                                         STATE$  ,&P$MOID
    748 000000                                          TRAN$   !MODID,         $EXIT
    749                                        ;
    750                                        ; NODE
    751                                        ;
    752 000000                                         STATE$  IDNOD
    753 000000                                          TRAN$  &K$EXE,          $EXIT,          $EXNOD
    754 000000                                          TRAN$  &K$NOD
    755 000000                                         STATE$  ,&P$NID
    756 000000                                          TRAN$  !NODID,          $EXIT,          $STEND
    757                                        ;
    758                                        ; OBJECT
    759                                        ;
    760 000000                                         STATE$  IDOBJ
    761 000000                                          TRAN$  &K$OBJ
    762 000000                                         STATE$  ,&P$OID
    763 000000                                          TRAN$  !OBJID,          $EXIT
    764                                        ;
    765                                        ; PROCESS
    766                                        ;
    767 000000                                         STATE$  IDPRO
    768 000000                                          TRAN$  &K$PRO
    769 000000                                         STATE$  ,&P$PID
    770 000000                                          TRAN$  !PROID,          $EXIT
    771                                        ;
    772                                        ; SYSTEM
    773                                        ;
    774 000000                                         STATE$  IDSYS
    775 000000                                          TRAN$  &K$SYS,          $EXIT
```

```
   1412                                       ;
   1413 000000                                        STATE$  OPUCSR
   1414 000000                                          TRAN$  &K$UNI,          OPUCS1
   1415 000000                                          TRAN$  $LAMDA
   1416 000000                                        STATE$  OPUCS1
   1417 000000                                          TRAN$  &K$CSR
   1418 000000                                        STATE$  ,&P$CSR
   1419 000000                                          TRAN$  $NUMBR,          $EXIT,     $SCSR,     LS.OPT,       RQBST
   1420                                       ;
   1421                                       ; USER
   1422                                       ;
   1423 000000                                        STATE$  OPUSE
   1424 000000                                          TRAN$  &K$USE
   1425 000000                                        STATE$  ,&P$OBUS
   1426 000000                                          TRAN$  &K$DEF,          $EXIT
   1427 000000                                          TRAN$  &K$LGI,          $EXIT,     ,          OF$LOG,       RQB+LO.OFL
   1428                                       ;
   1429                                       ; VECTOR
   1430                                       ;
   1431 000000                                        STATE$  OPVEC
   1432 000000                                          TRAN$  &K$VEC
   1433 000000                                        STATE$  ,&P$VEC
   1434 000000                                          TRAN$  $NUMBR,          $EXIT,     $VFC,      LS.OPT,       RQBST
   1435                                       ;
   1436                                       ; VERIFICATION (OBJECT)
   1437                                       ;
   1438 000000                                        STATE$  OPOBVE
   1439 000000                                          TRAN$  &K$VER,          ,          $OVER
   1440 000000                                        STATE$  ,&P$OBVE
   1441 000000                                          TRAN$  &K$ON,           $EXIT
   1442 000000                                          TRAN$  &K$OFF,          $EXIT,     ,          OF$OFF,       RQB+LO.OFL
   1443 000000                                          TRAN$  &K$INS,          $EXIT,     ,          OF$INS,       RQB+LO.OFL
   1444                                       ;
   1445                                       ; VERIFICATION STATE
   1446                                       ;
   1447 000000                                        STATE$  OPVER
   1448 000000                                          TRAN$  &K$VER
   1449 000000                                        STATE$
   1450 000000                                          TRAN$  &K$STAE,         OPVER1
   1451 000000                                          TRAN$  $LAMDA
   1452 000000                                        STATE$  OPVER1,&P$VEST
   1453 000000                                          TRAN$  &K$OFF,          $EXIT
   1454 000000                                          TRAN$  &K$ON,           $EXIT,     ,          1,            RQB+LO.VER
```

```
                                                                                                                   E 6
VSTB - VNP COMMAND PROCESSING S MACRO V05.03b Saturday 29-Jun-85 02:59  Page 22-3
Symbol table

SEMOX5  001770R     002 TIMER   006630R       002 $COP  = ****** G          $LTM  = ****** G          $STALI= ****** G
SENO    001530R     002 TY$CHA= 000400            $COST = ****** G          $MACT = ****** G          $STDES= ****** G
SENOOP  001542R     002 TY$EVE= 001000            $CSR  = ****** G          $MDEA = ****** G          $STEND= ****** G
SEOB    002006R     002 TY$STA= 002000            $CTL  = ****** G          $MXCO = ****** G          $STHOS= ****** G
SEOBOP  002032R     002 TY$SUM= 000000            $DACC = ****** G          $MXLI = ****** G          $STLND= ****** G
SEOB1   002020R     002 T$$KMG= 000000            $DIGIT= 000024            $NADD = ****** G          $STLSH= ****** G
SEPR    002102R     002 T$$MIN= 000000            $DNUMB= 000014            $NNAM = ****** G          $STLSN= ****** G
SEPROP  002132R     002 UID     004010R       002 $DPAS = ****** G          $NUMBR= 000002            $STNAD= ****** G
SEPR1   002120R     002 VF.ADD= ****** G          $DUSE = ****** G          $ONAM = ****** G          $STNAM= ****** G
SESY    002202R     002 VF.ECH= ****** G          $EOS  = 000012            $OPTON= ****** G          $STPAR= ****** G
SET     000630R     002 VF.FIX= ****** G          $EXIT = 000000            $OTYP = ****** G          $STRNG= 000004
SHO     002222R     002 VF.ON = ****** G          $EXNOD= ****** G          $OUT  = ****** G          $STRPA= ****** G
SHOALI  002770R     002 VF.PEM= ****** G          $FAIL = 177777            $OVER = ****** G          $STSNM= ****** G
SHOID   002276R     002 VF.PSI= ****** G          $FLAGS= ****** G          $OWN  = ****** G          $STTPA= ****** G
SHOLOG  002744R     002 VF.SED= ****** G          $GOFF = 000001            $PCIR = ****** G          $SUBXP= 000010
SHOOPT  002674R     002 V$$CTR= 001000            $GPRM = 000001            $PDT  = ****** G          $TRI  = ****** G
SHOTYP  002634R     002 WC.CNT= 000400            $GRO  = ****** G          $PNAM = ****** G          $TRIBA= ****** G
SHOXAC  003014R     002 WC.EVE= 004000            $HLPAC= ****** G          $PRGCL= ****** G          $TYPE = ****** G
SHOXPR  003034R     002 WC.TRI= 002000            $HTM  = ****** G          $PRI  = ****** G          $UNT  = ****** G
SHOXSV  003062R     002 WC.UNT= 001000            $IFALI= ****** G          $QUAL = ****** G          $VEC  = ****** G
SHO1    002244R     002 X$$DBT= 000000            $IFLOG= ****** G          $RAD50= 000016            $VFLAG= ****** G
SHO2    002256R     002 $ADS  = ****** G          $IFNNA= ****** G          $RET  = ****** G          $XMISS= ****** G
STMAT1  000020R         $ALPHA= 000022            $IFXAC= ****** G          $RONLY= 000001            $XTRI = ****** G
STMAT3  000030R         $ANT  = ****** G          $IFXPR= ****** G          $SCOND= ****** G          $XUNT = ****** G
STR     007042R     002 $ANY  = 000020            $IFX5S= ****** G          $SCOP1= ****** G          $$$FLG= 177777
STR.B   000134R         $ARAD1= ****** G          $IFX9S= ****** G          $SCOP2= ****** G          $$$KEY= 000154
STR.M   007060R     002 $ARAD2= ****** G          $INA  = ****** G          $SCOP3= ****** G          $$$LEN= 000007
S$$WRG= 000000          $BLANK= 000006            $INC  = ****** G          $SCSR = ****** G          $$$STA= 007060R      002
S$$YSZ= 007600          $CKLOG= ****** G          $LAMDA= 000000            $SEG  = ****** G          $$$TMP= 001413R      004
TAB   = 000011          $CMAND= ****** G          $LNAME= ****** G          $STADD= ****** G          .PFLAG= ****** G


. ABS.  000106    000   (RW,I,GBL,ABS,OVR)
        000162    001   (RW,I,LCL,REL,CON)
$STATE  007076    002   (RO,D,LCL,REL,CON)
$KTAB   000332    003   (RO,D,LCL,REL,CON)
$KSTR   001417    004   (RO,D,LCL,REL,CON)
$PRMPT  000000    005   (RO,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 15560 Words  ( 61 Pages)
Size of core pool: 16552 Words  ( 63 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:08:20.53
SY:VSTB.V2,[132,134]VSTB/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VSTB
```

```
                                                     F 6
VSTB       CREATED BY  MACRO  ON 29-JUN-85 AT 03:03      PAGE 1
```

Table of contents

```
    5-    54     MACRO DEFINITIONS
```

```
VSYM       CREATED BY  MACRO  ON 29-JUN-85 AT 03:07     PAGE 5
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

NHWDF$           #5-58        5-63
PDVDF$           #5-58        5-64
SLTDF$           #5-58        5-65
```

FILEID**VTMP

```
    405                                                  .SBTTL  TM.RD - READ RECORD
    406                                          ;+
    407                                          ; TM.RD - READ ONE RECORD FROM TEMPLATE FILE
    408                                          ;
    409                                          ; INPUTS:
    410                                          ;       NONE
    411                                          ;
    412                                          ; OUTPUTS:
    413                                          ;       C-BIT=SUCCESS/FAILURE
    414                                          ;       R0,R1,R2=DESTROYED
    415                                          ;       R3=RECORD LENGTH
    416                                          ;       R4=RECORD ADDRESS
    417                                          ;-
    418 001056  012767  000352'  000000G  TM.RD:  MOV     #TMPBF,$CLBUF   ; SET GCL BUFFER ADDRESS
    419 001064  012767  000204   000000G          MOV     #132.,$CLSIZ    ; AND BUFFER SIZE
    420 001072                                    CALL    $GCLEP          ; READ 1 RECORD
    421 001076  016700  000000G                   MOV     $CLIOS,R0       ; GET GCL IOSB ADDRESS
    422 001102  016003  000002                    MOV     2(R0),R3        ; GET RECORD LENGTH
    423 001106  010367  000556'                   MOV     R3,TMPSZ        ; SAVE IT
    424 001112  012704  000352'                   MOV     #TMPBF,R4       ; GET RECORD ADDRESS
    425 001116  105710                            TSTB    (R0)            ; SUCCESS? (CLEARS C-BIT)
    426 001120  100401                            BMI     101$            ; IF MI, NO
    427 001122                                    RETURN
    428
    429                                          ;
    430                                          ; ERROR CONDITIONS
    431                                          ;
    432 001124  122710  000000G           101$:   CMPB    #IE.EOF,(R0)    ; END OF FILE?
    433 001130  001405                            BEQ     111$            ; IF EQ, YES
    434 001132  122710  000000G                   CMPB    #IE.RBG,(R0)    ; BUFFER OVERFLOW?
    435 001136  001404                            BEQ     121$            ; IF EQ, YES
    436 001140                                    CALLR   ERR$10          ; READ FAILURE
    437 001144                            111$:   CALLR   ERR$11          ; NO END STATEMENT (WARNING)
    438 001150                            121$:   CALLR   ERR$12          ; READ OVERFLOW
```

```
VTMP       CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 4

MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              8-225    8-227    8-228    8-230    8-231    8-232    8-234    8-236    8-237    8-245
                  8-251    8-255    8-259    8-261    8-264    8-273    8-280    9-306    9-308    10-354
                  10-393   11-420   13-494   14-520   15-540   16-557   18-620   18-627   18-640   18-643
CALLR             8-274    8-281    11-436   11-437   11-438   12-453
EMSG$            #5-58     13-494   15-540   16-557
EMSG$R           #5-58     17-588   17-589
ERBLK$           #5-59     10-401
ERMSG$           #5-59     10-398   10-399   10-400
ERRPT$           #5-59     10-393
NTLER$           #5-58     7-194    7-195    7-       7-197    7-198
PUTAD            #5-58     9-306    10-354
PUTRC            #5-58
RAND             #9-306    9-306   #10-354   10-354
RETURN            8-240    8-267    9-309    10-374   11-427   13-503   14-525   15-541   16-558   17-583
                  18-645
SWTCHO           #5-58     8-239    9-305    10-341   10-373
```

```
VTMPIN     CREATED BY  MACRO  ON 29-JUN-85 AT 03:08      PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              8-131      8-147      8-159      8-170      8-173      8-177      8-178      9-208      9-212
EMSG$            #5-58       8-159      8-173
EMSG$R           #5-58       8-190
GETRV            #5-58       8-147      9-208      9-212
NKRDF$           #5-58       5-61
NTLER$           #5-58       7-86       7-87
RAND             #8-147      8-147     #9-208      9-208     #9-212      9-212
RESRG            #5-58       9-225
RETURN            8-181      9-229
SAVRG            #5-58       9-207
```

FILEID**VTST

```
Symbol table

ALL16   002710R           GETAD   001216R           M$$OVR= 000000            R$$11M= 000000            $CBTA = ****** GX
ALL22   003264R           GETRC   001134R           NEWLIN  002042R           SSTTBL  000030R           $CLBUF= ****** GX
A$$CHK= 000000            GETRV   001052R           NO    = 000000            SYSFDB= ****** GX         $CLIOS= ****** GX
A$$CPS= 000000            G$$TPP= 000000            NUL     002052R           S$$WRG= 000000            $CLPMT= ****** GX
A$$PRI= 000000            G$$TSS= 000000            N$$ACC= 000001            S$$YSZ= 007600            $COTB = ****** GX
A$$TRP= 000000            G$$TTK= 000000            N$$BUF= 000001            TAB     002044R           $CRAVL= ****** GX
BPT     001020R           G$$WRD= 000000            N$$LDV= 000001            TST     000036R           $DEA16= ****** GX
BUFOU   003640R           IO.WLB= ****** GX         N$$MCP= 000001            T$$KMG= 000000            $DEA22= ****** GX
CR    = 000053            I$$RAR= 000000            N$$MLL= 000001            T$$MIN= 000000            $DEVHD= ****** GX
CVBTO   004074R           I$$RDN= 000000            N$$MOV= 000010            VECTOR  000674R           $FMASK= ****** GX
C$$CKP= 000000            K$$CNT= 177546            N$$NCT= 000001            VF.INI= ****** GX         $GCLEP= ****** GX
C$$ORE= 000400            K$$CSR= 177546            N$$PEM= 000001            VF.VDT= ****** GX         $GETAD= ****** GX
C$$RSH= 177564            K$$LDC= 000000            N.FID = ****** GX         V$$CTR= 001000            $GETRC= ****** GX
DB    = 015012            K$$TPS= 000074            OB    = 015010            X$$DBT= 000000            $GETRV= ****** GX
DEA16   003154R           LD$LP = 000000            ODT     000764R           X10     002552R           $PARHD= ****** GX
DEA22   003530R           LF    = 000044            OW    = 031010            X101    001132R           $PUTAD= ****** GX
DELIM   002046R           LFCR  = 000040            PROMPT  004026R           X102    001214R           $PUTRC= ****** GX
DS    = 000060            L$$ASG= 000000            PUTAD   001326R           X103    001324R           $RADDR= ****** GX
DUMPB   001520R           L$$DRV= 000000            PUTRC   001436R           X104    001434R           $RBLCK= ****** GX
DUMPW   002054R           L$$P11= 000001            P$$P45= 000000            X105    001516R           $RSIZE= ****** GX
DW    = 031012            L$$11R= 000000            P$$WRD= 000000            X106    002772R           $VFLAG= ****** GX
D$$BUG= 177514            MADDR = 000014R           Q$$OPT= 000010            X107    003152R           $VIMI = ****** GX
D$$ISK= 000000            MCTL  = 000020R           Q.IOAE= 000012            X108    003262R           $VIMO = ****** GX
D$$L11= 000001            MLGTH = 000016R           Q.IOEF= 000006            X109    003346R           $VTST   000036RG
D$$YNC= 000000            MODIFY  002412R           Q.IOFN= 000002            X110    003526R           $$$   = 001656R      002
D$$YNM= 000000            MODPMT  002406R           Q.IOLU= 000004            X111    003636R           $$$ARG= 000003
EXIT    000066R           MSG     000000R           Q.IOPL= 000014            X20     002652R           $$$LEN= 000011
E$$XPR= 000000            M$$CRB= 000124            Q.IOPR= 000007            X30     002702R           $$$OST= 000014
FE.CEX= ****** GX         M$$CRX= 000000            Q.IOSB= 000010            X50     002706R           .CRAVL= ****** GX
F$$LVL= 000001            M$$FCS= 000000            R$$DER= 000000            $ALL16= ****** GX         .DEVHD= ****** GX
F.FNB = ****** GX         M$$MGE= 000000            R$$K11= 000001            $ALL22= ****** GX         .FMASK= ****** GX
F.NRBD= ****** GX         M$$NET= 000000            R$$SND= 000000            $BFR  = ****** GX         .PARHD= ****** GX
F.RSIZ= ****** GX

. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        004114    001   (RW,I,LCL,REL,CON)
$$$VDT  001670    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 11153 Words  ( 44 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:31.38
SY:VTST.V2,[132,134]VTST/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VTST
```

```
                                        .ENABL  LSB

                                ;
                                ; GET FIRST UMR BLOCK
                                ;
000304                          GFIRST: .IF     DF,R$$MPL                                               ;[MP01]
                                        TST     .MPRO           ; IS THIS A MULTIPROCESSOR SYSTEM?      ;[MP01]
                                        BLE     8$              ; BR IF NO                              ;[MP01]
                                        MOV     .UMRHD,R2       ; POINT AT THE FIRST UMR BLOCK          ;[MP01]
                                        MOV     $CPBIA,$RBLCK   ; SET THE CPU'S DATA AREA BIAS          ;[MP01]
                                        SWTCHI  #PREV           ; SET TO ''PREVIOUS UMR BLOCK'' BUFFER  ;[MP01]
                                        GETAD   R2,#M.LGTH      ; GET THE FIRST UMR BLOCK               ;[MP01]
                                        BR      9$              ; AND CONTINUE NORMALLY                 ;[MP01]
                                        .ENDC                                                           ;[MP01]

000304  016702  000000G         8$:     MOV     .UMRHD,R2       ; GET $UMRHD VALUE
000310  022702  177777                  CMP     #-1,R2          ; IS THE SMBOL DEFINED?
000314  000261                          SEC                     ; ASSUME FAILURE
000316  001445                          BEQ     20$             ; IF EQ, TARGET SYSTEM NOT 11/70!
000320                                  SWTCHI  #PREV           ; CHANGE DEFAULT BUFFER
000326                                  GETRV   R2,#M.LGTH      ; READ IN EXEC UMR BLOCK

000344                          9$:     SWTCHI  #NEXT           ; CHANGE DEFAULT BUFFER AGAIN
000352  000415                          BR      10$             ; JOIN COMMON CODE

                                ;
                                ; GET NEXT UMR BLOCK
                                ;
000354                          GNEXT:
000354  000261                          SEC                     ; ASSUME FAILURE
000356  010302                          MOV     R3,R2           ; COPY REAL MEMORY ADDRESS
000360  001424                          BEQ     20$             ; IF ZERO, END OF LIST (C-BIT SET)
000362  012704  000014'                 MOV     #NEXT,R4        ; COPY ''NEXT'' BLOCK
000366  012705  000000'                 MOV     #PREV,R5        ; TO ''PREVIOUS'' BLOCK
        000006                          .REPT   M.LGTH/2
                                        MOV     (R4)+,(R5)+
                                        .ENDR

                                ;
                                ; COMMON CODE
                                ;
000406                          10$:
000406  016703  000000'                 MOV     PREV,R3         ; GET NEXT POINTER FROM THIS BLOCK
000412  001406                          BEQ     15$             ; IF ZERO, THERE ARE NO OTHERS
000414  162703  000002                  SUB     #2,R3           ; GIVING START OF BLOCK ADDRESS

000420                                  GETRV   R3              ; READ IN NEXT UMR BLOCK
000430  000241                  15$:    CLC                     ; INDICATE SUCCESS
000432                          20$:    RETURN

                                        .DSABL  LSB

        000001                          .END
```

Table of contents

```
     5-    55     MACRO DEFINITIONS
     6-    92     LOCAL DATA
     7-   193     DEVEC - DEALLOCATE VECTOR BLOCK
     8-   241     $XPRO - UNSUCCESSFUL SET PROCESS CLEANUP
     9-   397     GETIND - GET PDV INDEX
    10-   426     DEAXPT - DEALLOCATE XPT DATA STRUCTURES
    11-   507     DEAECL - DEALLOCATE ECL NODE COUNTERS
    12-   537     $XLIN - UNSUCCESSFUL SET LINE CLEANUP
    13-   601     $CPRO - CLEAR PROCESS
    14-   630     $CLIN - CLEAR LIN
    15-   812     FNDADD - FIND LINE TABLE ADDRESS
    17-   948     VECT - RESET INTERRUPT VECTORS
    18-  1071     ICBSC - ICB SCAN
    19-  1169     ZDDS - CLEAR DDM LINE TABLE ADDRESS
    20-  1191     CSRUNL - REMOVE CSR UNLOAD ENTRY
    21-  1240     $CCEX - CLEAR SYS
    22-  1578     FLT16 - FIND LINE TABLE
    23-  1644     XLT16 - DEALLOCATE LINE TABLE
    24-  1783     .DEA16 - DEALLOCATE 16 BIT
    25-  1834     TSKSCH - SEARCH FOR TASK TCB
```

```
                                        .SBTTL  $XLIN - UNSUCCESSFUL SET LINE CLEANUP
                                ;+
                                ; $XLIN - CLEAN UP AFTER AN UNSUCCESSFUL "SET LINE"
                                ;
                                ; THIS ROUTINE IS CALLED IF A "SET LINE" FAILS IN TRANSFERRING EITHER THE
                                ; DLC OR DDM PROCESS INTO CORE, INITIALIZING THE DLC LINE TABLE, OR LOADING
                                ; THE DDM MICRO-CODE. IN NO CASE HAS THE DDM LINE TABLE BEEN INITIALIZED.
                                ;
                                ; INPUTS:
                                ;       $SLTA=SLT ADDRESS
                                ;       $DLCPV=DLC PDV ADDRESS
                                ;       $DDMPV=DDM PDV ADDRESS
                                ;       FL.PX1 OF $FLAGS SET IF DLC WAS TRANSFERRED
                                ;       FL.PX2 OF $FLAGS SET IF DDM WAS TRANSFERRED
                                ;
                                ; OUTPUTS:
                                ;       C-BIT=SET
                                ;-

001320                          $XLIN::
                                        .MCALL  VNPBPT
001320                                  VNPBPT  XLIN
001320  016705  000000G                 MOV     $SLTA,R5        ; GET SLT ADDRESS

001324  016500  000000G                 MOV     L.KRBA(R5),R0   ; GET THE KRB ADDRESS
001330  001404                          BEQ     5$              ; IF EQ, NONE
001332                                  CALL    XKRB            ; DEALLOCATE THE KRB
001336  005065  000000G                 CLR     L.KRBA(R5)      ; ZERO KRB ADDRESS

001342  016503  000000G         5$:     MOV     L.DLS(R5),R3    ; AND DLC LINE TABLE ADDRESS
001346  001417                          BEQ     20$             ; IF ZERO, NO WORK TO DO
                                ;
                                ; CHECK IF DLC WAS LOADED WITH THIS LINE
                                ;
001350  032767  000000G 000000G         BIT     #FL.PX1,$FLAGS  ; WAS DLC TRANSFERRED INTO CORE ?
001356  001003                          BNE     10$             ; IF NE, YES - DISCARD DLC LINE TABLES
001360  016767  000000G 000202'         MOV     $DLCPV,..PDV    ; ELSE .. HOUSEKEEP THE DE-ALLOCATION

001366  116504  000000G         10$:    MOVB    L.DLC(R5),R4    ; GET THE DLC PDV INDEX
001372                                  CALL    XLT16           ; DE-ALLOCATE DLC LINE TABLE
001376  005065  000000G                 CLR     L.DLS(R5)       ; ZERO LINE TABLE ADDRESS
001402  005065  000000G                 CLR     L.DLM(R5)       ; AND LINE TABLE BIAS

001406  032767  000000G 000000G 20$:    BIT     #FL.PX1,$FLAGS  ; WAS DLC TRANSFERRED?
001414  001405                          BEQ     30$             ; IF EQ, NO
001416  016704  000000G                 MOV     $DLCPV,R4       ; GET DLC PDV ADDRESS
001422  001415                          BEQ     101$            ; IF ZERO, LOGIC ERROR
001424                                  CALL    XDLC            ; RELEASE DLC FROM PARTITION
001430  032767  000000G 000000G 30$:    BIT     #FL.PX2,$FLAGS  ; WAS DDM TRANSFERRED?
001436  001405                          BEQ     40$             ; IF EQ, NO
001440  016704  000000G                 MOV     $DDMPV,R4       ; GET DDM PDV ADDRESS
001444  001405                          BEQ     111$            ; IF ZERO, LOGIC ERROR
001446                                  CALL    XPRO            ; RELEASE DDM FROM PARTITION
001452  000261                  40$:    SEC                     ; RETURN WITH C-BIT SET

001454                                  RETURN

```

```
   1062                                          ;
   1063 002732                                  101$:   CRASH                           ; LINE TABLE NOT FOUND WITHIN UNLOAD DATA
   1064 002734                                  111$:   CRASH                           ; NO VECTORS POINTED AT LINE TABLE!
   1065 002736                                  121$:   ERRPT$  ERR44,$EROUT    ; VECTOR SPACE EXCEEDS NOMINAL LIMITS
   1066 002746                                          ERMSG$  <The highest legal vector address in RSX11M systems is %n>
   1067 003036                                          ERMSG$  <1000, but in this system it is %O. Vector still in use.>
   1068 003125                                          ERBLK$  ERR44,<$MXVCT>
   1069
```

```
                                                                   F  1
VRQB - VNP PROCESS REQUEST BLOC MACRO V05.03b Tuesday 30-Jul-85 10:53  Page 35-1
Symbol table

AB    = ******  GX      ERRMSG  000370R    002  LO.COS  000020          LS.ALI= 000016          NULL    000234R    002
A$$CHK= 000000          E$$XPR= 000000          LO.CSR  000010          LS.CEX= 000001          N$$ACC= 000001
A$$CPS= 000000          FE.CEX= ******  GX      LO.CTL  000032          LS.CIR= 000033          N$$BUF= 000001
A$$PRI= 000000          FL.CEX= ******  GX      LO.DES  000002          LS.CX0= 010000          N$$LDV= 000001
A$$TRP= 000000          FL.DDM= ******  GX      LO.DTE  000047          LS.CX5= 000400          N$$MCP= 000001
CCEX    002536R         FL.DLC= ******  GX      LO.EVT  000004          LS.ECH= 001000          N$$MLL= 000001
CEXLD   000176R         FL.ECH= ******  GX      LO.GRD  000024          LS.FDX= 004000          N$$MOV= 000010
CHKLP   001264R         FL.FDX= ******  GX      LO.HOS  000040          LS.HDX= 010000          N$$NCT= 000001
CHKNDD  000234R         FL.HDX= ******  GX      LO.HTM  000032          LS.HLP= 000012          N$$PEM= 000001
CHKPP   000552R         FL.KMX= ******  GX      LO.INA  000062          LS.LIN= 000003          OF$INS= 000001
CLIN    002472R         FL.LLC= ******  GX      LO.INC  000060          LS.LMC= 000007          OF$LOG= 000100
CLQREM= ******  GX      FL.LMC= ******  GX      LO.LDT  000046          LS.LOG= 000020          OF$OFF= 000002
CLRPP   001532R         FL.MUX= ******  GX      LO.LID  000030          LS.MOD= 000036          OF$ON = 000000
CM$SET= ******  GX      FL.TOP= ******  GX      LO.LIN  000004          LS.NOD= 000014          OF$SMC= 000200
CPRD    001672R         FL.WRN= ******  GX      LO.LTM  000034          LS.NTI= 000013          OP$ADD= 000200
C$$CKP= 000000          FMAX  = 000037          LO.MAC  000022          LS.OBJ= 000022          OP$ALL= 000200
C$$DRE= 000400          FMT7    002352R    002  LO.MDE  000024          LS.OPT= 000400          OP$CCS= 000004
C$$RSH= 177564          FMT7A   002353R    002  LO.NAM  000044          LS.PRO= 000002          OP$CIE= 004000
DEQNTI  003640R         FMT7B   002366R    002  LO.NET= 000002          LS.TOP= 002000          OP$CON= 000400
DEV7    003424R         FM.7  = 000000          LO.NNM  000022          LS.TST= 000010          OP$COP= 000001
DSIZE = ******  GX      FM.7A = 000000          LO.OFL  000012          LS.UNF= 020000          OP$COS= 000100
DV.CCL= 000002          FM.7B = 000000          LO.ONM  000004          LS.XIT= 000011          OP$DST= 000200
DV.CDM= 020000          FTAB    000000R    002  LO.OUT  000056          LX.ALI= 000017          OP$DTE= 000040
DV.DIR= 000010          F$$LVL= 000001          LO.OWN  000026          LX.CEX= 000004          OP$DUP= 000002
DV.EXT= 000400          F.NRBD= ******  GX      LO.PAR  000010          LX.CIR= 000034          OP$EST= 000100
DV.F11= 040000          F.RSIZ= ******  GX      LO.PRI  000012          LX.LIN= 000006          OP$EVE= 010000
DV.ISP= 002000          F.URBD= ******  GX      LO.RET  000064          LX.LOG= 000021          OP$FIL= 000200
DV.MBC= 000400          G$$TPP= 000000          LO.RPA  000030          LX.NOD= 000015          OP$GRP= 000100
DV.MNT= 100000          G$$TSS= 000000          LO.SCR  000016          LX.OBJ= 000023          OP$HOS= 000001
DV.MSD= 000100          G$$TTK= 000000          LO.SEG  000054          LX.PRO= 000005          OP$HTM= 000002
DV.OSP= 004000          G$$WRD= 000000          LO.SKA  000014          L$$ASG= 000000          OP$INA= 004000
DV.PSE= 010000          I$$RAR= 000000          LO.SKN  000014          L$$DRV= 000000          OP$INC= 001000
DV.REC= 000001          I$$RDN= 000000          LO.SNM  000035          L$$P11= 000001          OP$KDS= 002000
DV.SDI= 000020          K$$CNT= 177546          LO.TPA  000020          L$$11R= 000000          OP$KDT= 000400
DV.SQD= 000040          K$$CSR= 177546          LO.TRB  000034          L.CTL = ******  GX      OP$KEV= 001000
DV.SWL= 001000          K$$LDC= 000000          LO.TRI  000030          L.DDM = ******  GX      OP$KGR= 001000
DV.TTY= 000004          K$$TPS= 000074          LO.UNT  000033          L.DDS = ******  GX      OP$KNT= 010000
DV.UMD= 000200          LBITS   000346R    002  LO.VCT  000014          L.DLC = ******  GX      OP$KSK= 002000
D$$BUG= 177514          LD$LP = 000000          LO.VER  000042          L.FLG = ******  GX      OP$LCT= 000200
D$$ISK= 000000          LD.ALI= 000030          LR.ACC  000020          L.NSTA= ******  GX      OP$LNE= 000040
D$$L11= 000001          LD.CEX= 000024          LR.ACD  000052          L.UNT = ******  GX      OP$LOC= 000001
D$$YNC= 000000          LD.CIR= 000035          LR.ADD  000012          MS.ALL= ******  GX      OP$LST= 000002
D$$YNM= 000000          LD.LIN= 000025          LR.ALI  000002          MS.CSR= ******  GX      OP$LTM= 000001
D.DSP   000012          LD.LOG= 000031          LR.BLK  000010          MS.EXT= ******  GX      OP$LVL= 000010
D.LNK   000000          LD.MOD= 000037          LR.CTL  000004          MS.INC= ******  GX      OP$MAC= 000010
D.MSK   000014          LD.NOD= 000027          LR.DES  000010          MS.LLC= ******  GX      OP$MCO= 000002
D.NAM   000004          LD.OBJ= 000032          LR.HLP  000002          MS.PRI= ******  GX      OP$MDE= 000400
D.PCB   000034          LD.PRO= 000026          LR.LIN  000002          MS.SEC= ******  GX      OP$MLI= 000004
D.UCB   000002          LF.ACT= ******  GX      LR.NAM  000012          MS.VCT= ******  GX      OP$MON= 000100
D.UCBL  000010          LF.RDY= ******  GX      LR.PAS  000041          M$$CRB= 000124          OP$MST= 000020
D.UNIT  000006          LF.UNL= ******  GX      LR.PRO  000002          M$$CRX= 000000          OP$NAM= 000004
D.VCAN= 000002          LLIN    000746R         LR.STS  000000          M$$FCS= 000000          OP$NET= 004000
D.VDEB= 177776          LO.ADD  000022          LR.TRI  000006          M$$MGE= 000000          OP$NLI= 000002
D.VINI= 000000          LO.ADR  000052          LR.TYP  000002          M$$NET= 000000          OP$NOD= 000020
D.VOUT= 000004          LO.CLS  000002          LR.UCB  000016          M$$OVR= 000000          OP$OUT= 002000
D.VPWF= 000006          LO.CON  000006          LR.UIC  000020          NOP     000120R         OP$OWN= 000020
ERRBUF  000400R    002  LO.COP  000010          LR.UNT  000005          NTINIT  002646R         OP$PAR= 000010
```

```
                                                                   G  1
VRQB - VNP PROCESS REQUEST BLOC MACRO V05.03b Tuesday 30-Jul-85 10:53  Page 35-2
```

```
**FILE**ID**VSET

VV        VV      SSSSSSSS      EEEEEEEEEE      TTTTTTTTTT
VV        VV      SSSSSSSS      EEEEEEEEEE      TTTTTTTTTT
VV        VV    SS              EE                   TT
VV        VV    SS              EE                   TT
VV        VV    SS              EE                   TT
VV        VV    SS              EE                   TT
VV        VV      SSSSSS        EEEEEEEE             TT
VV        VV      SSSSSS        EEEEEEEE             TT
VV        VV              SS    EE                   TT
VV        VV              SS    EE                   TT
   VV   VV                SS    EE                   TT             ....
   VV   VV                SS    EE                   TT             ....
     VV         SSSSSSSS        EEEEEEEEEE           TT             ....
     VV         SSSSSSSS        EEEEEEEEEE           TT             ....


LL                SSSSSSSS      TTTTTTTTTT
LL                SSSSSSSS      TTTTTTTTTT
LL              SS                   TT
LL              SS                   TT
LL              SS                   TT
LL              SS                   TT
LL                SSSSSS             TT
LL                SSSSSS             TT
LL                        SS         TT
LL                        SS         TT
LL                        SS         TT
LL                        SS         TT
LLLLLLLLLL      SSSSSSSS             TT
LLLLLLLLLL      SSSSSSSS             TT
```

```
    160 000000                          KEYWD$  ''LINE'',               K$LIN
    161 000000                          KEYWD$  ''LINES'',              K$LINS
    162 000000                          KEYWD$  ''LISTEN'',             K$LTM
    163 000000                          KEYWD$  ''LOAD'',               K$LOA
    164 000000                          KEYWD$  ''LOADER'',             K$LOAR
    165 000000                          KEYWD$  ''LOCATION'',           K$LOC
    166 000000                          KEYWD$  ''LOGIN'',              K$LGI
    167 000000                          KEYWD$  ''LOGGING'',            K$LOG
    168 000000                          KEYWD$  ''LOOP'',               K$LOO
    169 000000                          KEYWD$  ''MAXIMUM'',            K$MAX
    170 000000                          KEYWD$  ''MODULE'',             K$MOD
    171 000000                          KEYWD$  ''MONITOR'',            K$MON
    172 000000                          KEYWD$  ''MULTIPOINT'',         K$MUL
    173 000000                          KEYWD$  ''NAME'',               K$NAM
    174 000000                          KEYWD$  ''NETWORK'',            K$NET                       ;[TD001]
    175 000000                          KEYWD$  ''NETWORKS'',           K$NETS                      ;[TD001]
    176 000000                          KEYWD$  ''NODE'',               K$NOD
    177 000000                          KEYWD$  ''NODES'',              K$NODS
    178 000000                          KEYWD$  ''OBJECT'',             K$OBJ
    179 000000                          KEYWD$  ''OBJECTS'',            K$OBJS
    180 000000                          KEYWD$  ''OFF'',                K$OFF
    181 000000                          KEYWD$  ''ON'',                 K$ON
    182 000000                          KEYWD$  ''OUTGOING'',           K$OUT
    183 000000                          KEYWD$  ''OWNER'',              K$OWN
    184 000000                          KEYWD$  ''PARTITION'',          K$PAR
    185 000000                          KEYWD$  ''PASSWORD'',           K$PAS
    186 000000                          KEYWD$  ''PRIORITY'',           K$PRI
    187 000000                          KEYWD$  ''PROCESS'',            K$PRO
    188 000000                          KEYWD$  ''PROCESSES'',          K$PROS
    189 000000                          KEYWD$  ''PROTOCOL'',           K$PRTO
    190 000000                          KEYWD$  ''RECEIVE'',            K$REC
    191 000000                          KEYWD$  ''RETRANSMIT'',         K$RET
    192 000000                          KEYWD$  ''SCOPE'',              K$SCO
    193 000000                          KEYWD$  ''SECONDARY'',          K$SEC
    194 000000                          KEYWD$  ''SEGMENT'',            K$SEG
    195 000000                          KEYWD$  ''SERVER'',             K$SVR
    196 000000                          KEYWD$  ''SERVICE'',            K$SER
    197 000000                          KEYWD$  ''SET'',                K$SET
    198 000000                          KEYWD$  ''SHOW'',               K$SHO
    199 000000                          KEYWD$  ''SHUT'',               K$SHU
    200 000000                          KEYWD$  ''SINGLE'',             K$SNG
    201 000000                          KEYWD$  ''SINK'',               K$SNK
    202 000000                          KEYWD$  ''SINKS'',              K$SNKS
    203 000000                          KEYWD$  ''SIZE'',               K$SIZ
    204 000000                          KEYWD$  ''STATE'',              K$STAE
    205 000000                          KEYWD$  ''STATUS'',             K$STUS
    206 000000                          KEYWD$  ''SUMMARY'',            K$SUM
    207 000000                          KEYWD$  ''SYSTEM'',             K$SYS
    208 000000                          KEYWD$  ''TERMINAL'',           K$TER
    209 000000                          KEYWD$  ''TIMER'',              K$TIM
    210 000000                          KEYWD$  ''TOPDOWN'',            K$TOP
    211 000000                          KEYWD$  ''TRANSMIT'',           K$TRA
    212 000000                          KEYWD$  ''TRIBUTARY'',          K$TRIB
    213 000000                          KEYWD$  ''UNFIXED'',            K$UNF
    214 000000                          KEYWD$  ''UNIT'',               K$UNI
    215 000000                          KEYWD$  ''USER'',               K$USE
    216 000000                          KEYWD$  ''VDT'',                K$VDT
    777                                       .SBTTL ENTITY IDENTIFICATION SUBEXPRESSIONS
    778
    779                                       ;       Keyword         Next      Action     Mask (BIS)      Flag to
    780                                       ;       Type, or        State     Routine     Value           Mask
    781                                       ;       Substate
    782
    783                                 ;
    784                                 ; ENTITY IDENTIFICATION SUBEXPRESSIONS
    785                                 ;
    786                                 ; ALIAS ID
    787                                 ;
    788 000000                                STATE$  ALIID
    789 000000                                 TRAN$  $LAMDA,          ,        BL.MAN
    790 000000                                STATE$
    791 000000                                 TRAN$  $RAD50,          ,        $STALI
    792 000000                                STATE$
    793 000000                                 TRAN$  !NODEND,        ALIID1
    794 000000                                 TRAN$  $LAMDA
    795 000000                                STATE$  ALIID1
    796 000000                                 TRAN$  $LAMDA,         $EXIT,    BL.AUT
    797                                 ;
    798                                 ; CIRCUIT ID
    799                                 ;
    800 000000                                STATE$  CIRID
    801 000000                                 TRAN$  $RAD50,          ,        $LNAME
    802 000000                                STATE$
    803 000000                                 TRAN$  <'->
    804 000000                                STATE$
    805 000000                                 TRAN$  $DNUMB,         CIR1,     $CTL
    806 000000                                 TRAN$  <'*>,            ,           ,        WC.CNT,          $QUAL
    807 000000                                STATE$  CIR1
    808 000000                                 TRAN$  <'->,           CIR2
    809 000000                                 TRAN$  <'.>,           CIR4
    810 000000                                 TRAN$  $LAMDA,         $EXIT,       ,        FL.CNT,          $FLAGS
    811 000000                                STATE$  CIR2
    812 000000                                 TRAN$  $DNUMB,         CIR3,     $UNT
    813 000000                                 TRAN$  <'*>,           CIR3,     $XUNT,      WC.UNT,          $QUAL
    814 000000                                STATE$  CIR3
    815 000000                                 TRAN$  <'.>,           CIR4
    816 000000                                 TRAN$  $LAMDA,         $EXIT
    817 000000                                STATE$  CIR4
    818 000000                                 TRAN$  $DNUMB,         $EXIT,    $TRI
    819 000000                                 TRAN$  <'*>,           $EXIT,    $XTRI,      WC.TRI,          $QUAL
    820 000000                                 TRAN$  $LAMDA,         $EXIT
    821                                 ;
    822                                 ; DESTINATION NODE ID
    823                                 ;
    824 000000                                STATE$  DESID
    825 000000                                 TRAN$  !DNODID
    826                                 ;
    827                                 ;     DESTINATION NODE, RSX FORMAT
    828                                 ;
    829 000000                                STATE$
    830 000000                                 TRAN$  '/,             UID
    831 000000                                 TRAN$  $LAMDA,         DESIR1
    832 000000                                STATE$  UID
    833 000000                                 TRAN$  !DSTR,           ,        $DUSE
```

```
   1456                                         .SBTTL  OPTION KEYWORD AND VALUE SUBEXPRESSIONS
   1457
   1458                                         ;       Keyword         Next      Action    Mask (BIS)      Flag to
   1459                                         ;       Type, or        State     Routine     Value           Mask
   1460                                         ;       Substate
   1461
   1462                                 ;
   1463                                 ; OPTION KEYWORD SUBEXPRESSIONS
   1464                                 ;
   1465                                 ; COUNTER TIMER
   1466                                 ;
   1467 000000                                  STATE$  COUTIM
   1468 000000                                   TRAN$   &K$COU
   1469 000000                                  STATE$  TIMER
   1470 000000                                   TRAN$   &K$TIM,        $EXIT
   1471                                 ;
   1472                                 ; HELLO TIMER
   1473                                 ;
   1474 000000                                  STATE$  HELTIM
   1475 000000                                   TRAN$   &K$HTM,        TIMER
   1476                                 ;
   1477                                 ; LISTEN TIMER
   1478                                 ;
   1479 000000                                  STATE$  LISTIM
   1480 000000                                   TRAN$   &K$LTM,        TIMER
   1481                                 ;
   1482                                 ; RECEIVE PASSWORD
   1483                                 ;
   1484 000000                                  STATE$  KWRPAS
   1485 000000                                   TRAN$  &K$REC
   1486 000000                                  STATE$
   1487 000000                                   TRAN$  &K$PAS,         $EXIT
   1488 000000                                   TRAN$  $LAMDA,         $EXIT
   1489                                 ;
   1490                                 ; TRANSMIT PASSWORD
   1491                                 ;
   1492 000000                                  STATE$  KWTPAS
   1493 000000                                   TRAN$  &K$TRA
   1494 000000                                  STATE$
   1495 000000                                   TRAN$  &K$PAS,         $EXIT
   1496 000000                                   TRAN$  $LAMDA,         $EXIT
   1497                                 ;
   1498                                 ; OPTION VALUE SUBEXPRESSIONS
   1499                                 ;
   1500                                 ; EVENT IDENTIFICATION
   1501                                 ;
   1502 000000                                  STATE$  EVEID
   1503 000000                                   TRAN$  $LAMDA,         ,         BL.MAN
   1504 000000                                  STATE$
   1505 000000                                   TRAN$  $DNUMB,         ,         CKEVEC
   1506 000000                                  STATE$
   1507 000000                                   TRAN$  <'.>
   1508 000000                                  STATE$
   1509 000000                                   TRAN$  <'*>,           EVEID2,           ,         WC.EVE,         $QUAL
   1510 000000                                   TRAN$  $LAMDA
   1511 000000                                  STATE$
   1512 000000                                   TRAN$  !EVELIS
```

```
                                                          F  6
VSTB       CREATED BY  MACRO  ON 29-JUN-85 AT 03:03     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE              REFERENCES

BL.AUT     000010 R        #17-1602
BL.MAN     000000 R        #17-1595
CKEVEB  =  ******   G        5-101
CKEVEC  =  ******   G        5-101
CKEVEN  =  ******   G        5-101
CKEVER  =  ******   G        5-101
CM$CLE  =  ******   G        5-85
CM$SET  =  ******   G        5-85
CM$SHO  =  ******   G        5-85
DSTR.E     000062 R        #20-1650
FL.CNT  =  ******   G        5-86
KEYTAB     000000 RG       #6-108
LD.ALI  =  000030          #5-62
LD.CEX  =  000024          #5-62
LD.CIR  =  000035          #5-62
LD.LIN  =  000025          #5-62
LD.LOG  =  000031          #5-62
LD.MOD  =  000037          #5-62
LD.NOD  =  000027          #5-62
LD.OBJ  =  000032          #5-62
LD.PRO  =  000026          #5-62
LO.ADD     000022          #5-62
LO.ADR     000052          #5-62
LO.CLS     000002          #5-62
LO.CON     000006          #5-62
LO.COP     000010          #5-62
LO.COS     000020          #5-62
LO.CSR     000010          #5-62          5-62
LO.CTL     000032          #5-62
LO.DES     000002          #5-62          5-62
LO.DTE     000047          #5-62
LO.EVT     000004          #5-62
LO.GRO     000024          #5-62
LO.HOS     000040          #5-62
LO.HTM     000032          #5-62
LO.INA     000062          #5-62
LO.INC     000060          #5-62
LO.LDT     000046          #5-62
LO.LID     000030          #5-62
LO.LIN     000004          #5-62
LO.LTM     000034          #5-62
LO.MAC     000022          #5-62
LO.MDE     000024          #5-62
LO.NAM     000044          #5-62
LO.NET  =  000002          #5-62
LO.NNM     000022          #5-62
LO.OFL     000012          #5-62
LO.ONM     000004          #5-62
LO.OUT     000056          #5-62
LO.OWN     000026          #5-62
LO.PAR     000010          #5-62
LO.PRI     000012          #5-62
```

```
                                                          G  6
VSTB       CREATED BY  MACRO  ON 29-JUN-85 AT 03:03     PAGE 2
```

```
                .TITLE  VSYM - VNP SYMBOL DEFINITIONS
                .IDENT  /V05.00/

        ;
        ; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
        ;
        ;
        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
        ; TRANSFERRED.
        ;
        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
        ; CORPORATION.
        ;
        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
        ;
        ; MODULE DESCRIPTION:
        ;
        ;       VNP - GLOBAL NTL SYMBOLS
        ;
        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
        ;
        ; IDENT HISTORY:
        ;
        ; 1.00  27-FEB-78
        ;       VERSION 2.0 RELEASE
        ;
        ; 2.00  14-DEC-79
        ;       DECNET-11M/S V3.0
        ;       DECNET-11M-PLUS V1.0
        ;
        ; 3.00  16-APR-82
        ;       DECNET-11M V3.1
        ;       DECNET-11M-PLUS V1.1
        ;
        ; 4.00  07-NOV-83
        ;       DECNET-11M V4.0
        ;       DECNET-11M-PLUS V2.0
        ;
        ; 5.00  22-JUL-85
        ;       DECnet-11M/S V4.2
        ;       DECnet-11M-Plus V3.0
        ;       DECnet-Micro/RSX V1.0
        ;
        ;
```

```
**FILE**ID**VTMP




VV      VV   TTTTTTTTTT   MM      MM   PPPPPPPP
VV      VV   TTTTTTTTTT   MM      MM   PPPPPPPP
VV      VV       TT       MMMM  MMMM   PP      PP
VV      VV       TT       MMMM  MMMM   PP      PP
VV      VV       TT       MM  MM  MM   PP      PP
VV      VV       TT       MM  MM  MM   PP      PP
VV      VV       TT       MM      MM   PPPPPPPP
VV      VV       TT       MM      MM   PPPPPPPP
VV      VV       TT       MM      MM   PP
VV      VV       TT       MM      MM   PP
  VV  VV         TT       MM      MM   PP                ....
  VV  VV         TT       MM      MM   PP                ....
    VV           TT       MM      MM   PP                ....
    VV           TT       MM      MM   PP                ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
    440                                               .SBTTL  $ELINE - ECHO LINE
    441                                       ;+
    442                                       ; $ELINE - ECHO LAST LINE READ FROM TEMPLATE FILE
    443                                       ;
    444                                       ; INPUTS:
    445                                       ;       NONE
    446                                       ;
    447                                       ; OUTPUTS:
    448                                       ;       R0,R1=DESTROYED
    449                                       ;-
    450 001154                                $ELINE::
    451 001154  012700  000352'                       MOV     #TMPBF,R0               ; GET BUFFER ADDRESS
    452 001160  016701  000556'                       MOV     TMPSZ,R1                ; AND LENGTH
    453 001164                                        CALLR   $TIOUT                  ; PRINT IT ON TI:
```

```
**FILE**ID**VTMPIN

VV      VV   TTTTTTTTTT   MM      MM   PPPPPPPP       IIIIII     NN      NN
VV      VV   TTTTTTTTTT   MM      MM   PPPPPPPP       IIIIII     NN      NN
VV      VV       TT       MMMM  MMMM   PP      PP       II       NN      NN
VV      VV       TT       MMMM  MMMM   PP      PP       II       NN      NN
VV      VV       TT       MM  MM  MM   PP      PP       II       NNNN    NN
VV      VV       TT       MM  MM  MM   PP      PP       II       NNNN    NN
VV      VV       TT       MM      MM   PPPPPPPP         II       NN  NN  NN
VV      VV       TT       MM      MM   PPPPPPPP         II       NN  NN  NN
VV      VV       TT       MM      MM   PP               II       NN    NNNN
VV      VV       TT       MM      MM   PP               II       NN    NNNN
  VV  VV         TT       MM      MM   PP               II       NN      NN      ....
  VV  VV         TT       MM      MM   PP               II       NN      NN      ....
    VV           TT       MM      MM   PP             IIIIII     NN      NN      ....
    VV           TT       MM      MM   PP             IIIIII     NN      NN      ....


LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

G 10

VTMPIN - VNP TEMPLATE INITIALIZ MACRO V05.03b Saturday 29-Jun-85 03:08

```
**FILE**ID**VTST






VV      VV    TTTTTTTTTT      SSSSSSSS   TTTTTTTTTT
VV      VV    TTTTTTTTTT      SSSSSSSS   TTTTTTTTTT
VV      VV        TT        SS               TT
VV      VV        TT        SS               TT
VV      VV        TT        SS               TT
VV      VV        TT        SS               TT
VV      VV        TT          SSSSSS         TT
VV      VV        TT          SSSSSS         TT
VV      VV        TT                SS       TT
VV      VV        TT                SS       TT
  VV  VV          TT                SS       TT            ....
  VV  VV          TT                SS       TT            ....
    VV            TT        SSSSSSSS         TT            ....
    VV            TT        SSSSSSSS         TT            ....




LL              SSSSSSSS   TTTTTTTTTT
LL              SSSSSSSS   TTTTTTTTTT
LL            SS               TT
LL            SS               TT
LL            SS               TT
LL            SS               TT
LL              SSSSSS         TT
LL              SSSSSS         TT
LL                    SS       TT
LL                    SS       TT
LL                    SS       TT
LL                    SS       TT
LLLLLLLLLL    SSSSSSSS         TT
LLLLLLLLLL    SSSSSSSS         TT
```

```
                                                                   F 12
VTST          CREATED BY  MACRO  ON 29-JUN-85 AT 03:09         PAGE 1
SYMBOL CROSS REFERENCE                                         CREF    04.00

SYMBOL   VALUE            REFERENCES

ALL16       002710 R        6-243     #6-389
ALL22       003264 R        6-245     #6-414
BPT         001020 R        5-169     #6-262
BUFOU       003640 R        6-188     #6-438
CR        = 000053         #5-69       6-307      6-321      6-339      6-353      6-368      6-397      6-399      6-422
                            6-424
CVBTO       004074 R        6-306      6-313      6-338      6-345      6-367      6-370      6-397      6-398      6-422
                            6-423     #6-481
DB        = 015012         #5-72
DEA16       003154 R        6-247     #6-404
DEA22       003530 R        6-249     #6-429
DELIM       002046 R       #6-327
DS        = 000060         #5-68       6-210      6-211
DUMPB       001520 R        6-237     #6-305
DUMPW       002054 R        6-239     #6-334
DW        = 031012         #5-74       6-398      6-423
EXIT        000066 R        6-185     #6-190
FE.CEX    = ******   GX     6-191      6-455
F.FNB     = ******   GX     6-439
F.NRBD    = ******   GX    *6-446     *6-448     *6-450     *6-452     *6-454
F.RSIZ    = ******   GX    *6-447
GETAD       001216 R        6-231     #6-281
GETRC       001134 R        6-229     #6-274
GETRV       001052 R        6-227     #6-267
IO.WLB    = ******   GX     5-59
LF        = 000044         #5-67       6-306      6-338      6-367      6-370      6-396      6-398      6-421      6-423
LFCR      = 000040         #5-66       6-212      6-213      6-214      6-215      6-216      6-217      6-218      6-219
                            6-220      6-221      6-222      6-223      6-262      6-308      6-340      6-362      6-382
                            6-392      6-409      6-417      6-434
MADDR     = 000014 R       #5-60      *6-210     *6-211     *6-212     *6-213     *6-214     *6-215     *6-216     *6-217
                           *6-218     *6-219     *6-220     *6-221     *6-222     *6-223     *6-262     *6-306     *6-307
                           *6-308     *6-313     *6-314     *6-319     *6-321     *6-338     *6-339     *6-340     *6-345
                           *6-346     *6-351     *6-353     *6-362     *6-367     *6-368     *6-370     *6-382     *6-392
                           *6-396     *6-397     *6-398     *6-399     *6-409     *6-417     *6-421     *6-422     *6-423
                           *6-424     *6-434
MCTL      = 000020 R       #5-62      *6-210     *6-211     *6-212     *6-213     *6-214     *6-215     *6-216     *6-217
                           *6-218     *6-219     *6-220     *6-221     *6-222     *6-223     *6-262     *6-306     *6-307
                           *6-308     *6-313     *6-314     *6-319     *6-321     *6-338     *6-339     *6-340     *6-345
                           *6-346     *6-351     *6-353     *6-362     *6-367     *6-368     *6-370     *6-382     *6-392
                           *6-396     *6-397     *6-398     *6-399     *6-409     *6-417     *6-421     *6-422     *6-423
                           *6-424     *6-434
MLGTH     = 000016 R       #5-61      *6-210     *6-211     *6-212     *6-213     *6-214     *6-215     *6-216     *6-217
                           *6-218     *6-219     *6-220     *6-221     *6-222     *6-223     *6-262     *6-306     *6-307
                           *6-308     *6-313     *6-314     *6-319     *6-321     *6-338     *6-339     *6-340     *6-345
                           *6-346     *6-351     *6-353     *6-362     *6-367     *6-368     *6-370     *6-382     *6-392
                           *6-396     *6-397     *6-398     *6-399     *6-409     *6-417     *6-421     *6-422     *6-423
                           *6-424     *6-434
MODIFY      002412 R        6-241     #6-362
MODPMT      002406 R       #6-359      6-371
MSG         000000 R       #5-59       5-60       5-61       5-62       6-210      6-211      6-212      6-213      6-214
                            6-215      6-216      6-217      6-218      6-219      6-220      6-221      6-222      6-223
                            6-262      6-306      6-307      6-308      6-313      6-314      6-319      6-321      6-338
```

```
                                                                   G 12
VTST          CREATED BY  MACRO  ON 29-JUN-85 AT 03:09         PAGE 2
```

```
VUMR - VNP DYNAMIC UMR ALLOC/DE MACRO V05.03b Saturday 29-Jun-85 03:09  Page 10-1
Symbol table

A$$CHK= 000000       G$$TSS= 000000       M.BFVH   000011          Q$$OPT= 000010         S.LHD    000000
A$$CPS= 000000       G$$TTK= 000000       M.BFVL   000012          R$$DER= 000000         S.PKT    000014
A$$PRI= 000000       G$$WRD= 000000       M.LGTH=  000014          R$$K11= 000001         S.PRI    000004
A$$TRP= 000000       I$$RAR= 000000       M.LNK    000000          R$$SND= 000000         S.RCNT   177772
C$$CKP= 000000       I$$RDN= 000000       M.UMRA   000002          R$$11M= 000000         S.ROFF   177773
C$$ORE= 000400       K$$CNT= 177546       M.UMRN   000004          SPARE  = 000010        S.STS    000011
C$$RSH= 177564       K$$CSR= 177546       M.UMVH   000010          SP.EIP= 000001         S.VCT    000005
D$$BUG= 177514       K$$LDC= 000000       M.UMVL   000006          SP.ENB= 000002         T$$KMG= 000000
D$$ISK= 000000       K$$TPS= 000074       NEXT     000014R     002 SP.LOG= 000004         T$$MIN= 000000
D$$L11= 000001       LD$LP  = 000000      N$$ACC=  000001          SYSFDB= ****** GX      UBMPR  = 170200
D$$YNC= 000000       L$$ASG= 000000       N$$BUF=  000001          S$$WRG= 000000         V$$CTR= 001000
D$$YNM= 000000       L$$DRV= 000000       N$$LDV=  000001          S$$YSZ= 007600         X$$DBT= 000000
E$$XPR= 000000       L$$P11= 000001       N$$MCP=  000001          S.BMSK   177776        $ASUMR   000000RG
F$$LVL= 000001       L$$11R= 000000       N$$MLL=  000001          S.BMSV   177774        $BFR   = ****** GX
F.NRBD= ****** GX    M$$CRB= 000124       N$$MOV=  000010          S.CON    000010        $DEUMR   000172RG
F.RSIZ= ****** GX    M$$CRX= 000000       N$$NCT=  000001          S.CSR    000012        $GETRV= ****** GX
F.URBD= ****** GX    M$$FCS= 000000       N$$PEM=  000001          S.CTM    000006        $PUTRC= ****** GX
GFIRST  000304R      M$$MGE= 000000       PREV     000000R     002 S.DMCS   000020        $RADDR= ****** GX
GNEXT   000354R      M$$NET= 000000       P$$P45=  000000          S.FRK    000016        $SAVAL= ****** GX
G$$TPP= 000000       M$$OVR= 000000       P$$WRD=  000000          S.ITM    000007        .UMRHD= ****** GX


. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000434    001   (RW,I,LCL,REL,CON)
DATA    000030    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 9833 Words  ( 39 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:10.42
SY:VUMR.V2,[132,134]VUMR/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VUMR
```

```
VUMR        CREATED BY  MACRO  ON 29-JUN-85 AT 03:09      PAGE 1
```

```
        .TITLE  VUNL - VNP UNLOAD ROUTINES
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       VNP - "CLEAR CEX"/"CLEAR LINE"/"CLEAR PROCESS" ROUTINES
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
    594                                     ;
    595                                     ; ERROR CONDITIONS
    596                                     ;
    597 001456                              101$:   CRASH                   ; NETLDR LOGIC ERROR
    598 001460                              111$:   CRASH                   ; DITTO
    599
```

```
   1071                                                 .SBTTL  ICBSC - ICB SCAN
   1072                                         ;+
   1073                                         ;       ICBSC - DETERMINE IF VECTOR POINTS AT LINE TABLE OR ICB
   1074                                         ;
   1075                                         ; INPUTS:
   1076                                         ;       R0 = VECTOR ADDRESS
   1077                                         ;       R1 = UPPER OF VECTOR ADDRESSES
   1078                                         ;       R2 = LINE TABLE ADDRESS
   1079                                         ;       R3 = UPPER BOUND OF LINE TABLE ADDRESS
   1080                                         ;       $ICBLN = SIZE OF EACH ICB
   1081                                         ;       ..PDV = PDV OF PROCESS
   1082                                         ;
   1083                                         ; OUTPUTS:
   1084                                         ;       R0 = VECTOR ADDRESS, ON A MATCH
   1085                                         ;       ICBSZ = INCREMENTED ON A MATCH
   1086                                         ;
   1087                                         ;-
   1088
   1089 003132                                  ICBSC:  SWTCHI  #ICBLK           ; PREPARE TO READ ICB
   1090 003140  016767  000000G 000000G                 MOV     $ICBLN,$RSIZE    ; SIZE OF A SINGLE ICB
   1091
   1092                                                 .IF NDF R$$MPL                                                   ;[MP01]
   1093 003146  005077  000000G                         CLR     @$ADRX           ; BIAS ($RBLCK)
   1094
   1095                                                 .ENDC                                                            ;[MP01]
   1096
   1097 003152  132767  000001  000200' 5$:     BITB    #PR.LTA,PRLTA    ; PARTITION LINE TABLE?
   1098 003160  001015                                  BNE     30$              ; YES IF NE
   1099 003162  021002                          10$:    CMP     (R0),R2          ; DOES VECTOR POINT AT LINE TABLE?
   1100 003164  103402                                  BLO     20$              ; NO IF LO
   1101 003166  021003                                  CMP     (R0),R3          ; WITHIN RANGE THEN?
   1102 003170  103462                                  BLO     36$              ; YES IF LO...SO GOT ONE
   1103 003172  132767  000002  000200' 20$:    BITB    #PR.SEC,PRLTA    ; BEEN THRU VECSC BEFORE?
   1104 003200  001060                                  BNE     40$              ; IF NE, YES...SO DONE
   1105 003202  022020                                  CMP     (R0)+,(R0)+      ; ADVANCE TO NEXT VECTOR
   1106 003204  020001                                  CMP     R0,R1            ; DONE YET?
   1107 003206  103761                                  BLO     5$               ; NO IF LOW
   1108 003210  005000                                  CLR     R0               ; INDICATE NO VECTORS FOUND
   1109 003212  000453                                  BR      40$              ; C-SET IF DIDN'T FIND ONE
   1110
   1111 003214  011004                          30$:    MOV     (R0),R4          ; COULD BE START OF ICB
   1112 003216  032704  000001                          BIT     #1,R4            ; MUST BE ON WORD BOUNDARY
   1113 003222  001363                                  BNE     20$              ; NOT IF NE
   1114 003224  020427  120000                          CMP     R4,#120000       ; MAPPED SO MUST BE LT APR5 ADDRESS
   1115 003230  103360                                  BHIS    20$              ; NOT IF HIS
   1116 003232  010467  000000G                         MOV     R4,$RADDR        ; VIRTUAL ADDRESS ($RADDR)
   1117 003236                                          GETAD                    ; READ
   1118 003242  012704  000150'                         MOV     #ICBLK,R4        ; PREPARE TO SCAN DATA
   1119 003246  022427  013746                          CMP     (R4)+,#13746     ; IS IT 'MOV'?
   1120 003252  001347                                  BNE     20$              ; NO IF NE
   1121
   1122                                                 .IF DF  R$$MPL                                                   ;[MP01]
   1123
   1124                                                 BIT     #DF.DAS,$DFLAG   ; D-SPACE ENABLED                       ;[MP01]
   1125                                                 BEQ     35$              ; IF NO - BRANCH                        ;[MP01]
   1126                                                 CMP     (R4)+,#172372    ; CHECK IF ITS OUR ICB                  ;[MP01]
   1127                                                 BNE     20$              ; IF NO - BRANCH                        ;[MP01]
```

```
Symbol table

OP$PRI= 000010          S1.RNF= 020000    S5.OXF= 000040    T2.CHK= 020000        US.RED= 000002
OP$RET= 010000          S1.RSP= 000004    S5.RPO= 002000    T2.CKD= 010000        US.SHR= 000001
OP$RPA= 000010          S1.RST= 000001    S5.SW1= 000001    T2.DST= 040000        US.SPU= 000002
OP$SCO= 000001          S1.TNE= 040000    S5.TMM= 000002    T2.FXD= 002000        US.UMD= 000010
OP$SEG= 000400          S1.TSY= 000040    S5.VER= 010000    T2.HLT= 000200        US.VV = 000001
OP$SER= 000400          S1.USI= 100000    S5.XOF= 000004    T2.REX= 001000        US.WCK= 000010
OP$SIN= 000010          S2.BEL= 020000    S5.XON= 000010    T2.SEF= 004000        UU.ABO= 000400
OP$STA= 000004          S2.BRQ= 000020    S6.EIO= 000400    T2.SPN= 000004        UU.ATN= 000100
OP$TPA= 000020          S2.CR = 002000    S6.RDI= 100000    T2.STP= 000020        UU.AVN= 000004
OP$TRI= 000040          S2.CTO= 040000    S6.RLU= 001000    T2.WFR= 000001        UU.GUS= 000010
OP$UCS= 000020          S2.CTS= 100000    TSKSCH  002652R   T3.ACP= 100000        UU.IOS= 002000
OP$USE= 000002          S2.ELF= 001000    TS.BLK= 171700    T3.CAL= 000100        UU.ONL= 000020
OP$VEC= 000040          S2.FLF= 000400    TS.CKP= 000200    T3.CLI= 001000        UU.RCT= 000002
OP$VER= 000040          S2.IRQ= 000200    TS.CKR= 000100    T3.GFL= 000010        UU.RDY= 000200
OP$XAC= 000001          S2.OBF= 004000    TS.EXE= 100000    T3.MCR= 004000        UU.SER= 000001
OP$XPR= 000002          S2.ORQ= 000100    TS.HLD= 002000    T3.NET= 000020        UU.SIO= 001000
OP$X5S= 000004          S2.PCU= 010000    TS.MSG= 020000    T3.NSD= 000200        UU.SPC= 000040
OP$X9S= 000010          S2.RCU= 000001    TS.NRP= 010000    T3.PMD= 040000        U.ACB   000062
PBITS   000356R     002 S2.SRQ= 000040    TS.OUT= 000400    T3.PRV= 010000        U.ACP = 000032
PDVA    000474R         S2.WAL= 000010    TS.RDN= 040000    T3.REM= 020000        U.ADMA  000066
PDVX    000246R     002 S2.WRA= 000006    TS.RUN= 004000    T3.ROV= 000040        U.AFLG  000064
PSILD   000216R         S2.WRB= 000002    TS.STP= 001000    T3.RST= 000400        U.ATT   000022
P$$P45= 000000          S3.ACR= 000001    TY$CHA= 000400    T3.SLV= 002000        U.BPKT= 000044
P$$WRD= 000000          S3.CTC= 000004    TY$EVE= 001000    T3.SWS= 000002        U.BUF   000024
QF$ACT= 000001          S3.FDX= 000200    TY$STA= 002000    UCBBUF  000252R   002 U.CBF = 000032
QF$ALL= 000002          S3.ICE= 001000    TY$SUM= 000000    UC.ALG= 000200        U.CLI   177772
QF$KNO= 000004          S3.MHE= 000400    T$$KMG= 000000    UC.ATT= 000010        U.CNT   000030
QF$LOO= 000010          S3.NEC= 000020    T$$MIN= 000000    UC.KIL= 000004        U.COTQ  000030
Q$$OPT= 000010          S3.PPT= 020000    T.ACTL  000052    UC.LGH= 000003        U.CTCB  000026
REP7    003344R         S3.PTH= 004000    T.ASTL  000016    UC.NPR= 000100        U.CTL   000004
REP7A   003346R         S3.RAL= 000010    T.ATT   000062    UC.PWF= 000020        U.CTYP  000052
REP7B   003456R         S3.RES= 010000    T.CPCB  000004    UC.QUE= 000040        U.CW1   000010
RNTI    000162R         S3.RUB= 040000    T.DPRI  000040    UD.UNS= 000000        U.CW2   000012
RQB   = ****** GX       S3.TAB= 000002    T.EFLG  000022    UD.160= 000004        U.CW3   000014
R$$DER= 000000          S3.TME= 002000    T.IOC   000003    UD.200= 000001        U.CW4   000016
R$$K11= 000001          S3.TSY= 000040    T.LBN   000041    UD.556= 000002        U.CYL = 000104
R$$SND= 000000          S3.8BC= 000100    T.LDV   000044    UD.625= 000005        U.DCB   000000
R$$11M= 000000          S4.ABD= 100000    T.LNK   000000    UD.8K = 000006        U.FCDE= 000042
SCEX    000124R         S4.ANI= 000400    T.MXSZ  000050    UD.800= 000003        U.FNUM= 000040
SCIR    001640R         S4.AVO= 001000    T.NAM   000006    UM.CLI= 000036        U.GRP = 000102
SLIN    001544R         S4.BLK= 002000    T.OFF   000066    UM.DSB= 000200        U.KCSR= 000032
SLTA    000774R         S4.DEC= 004000    T.PCB   000046    UM.NBR= 000400        U.KCS6= 000034
SPRO    000670R         S4.DLO= 000100    T.PRI   000002    UM.OVR= 000001        U.LUIC  177774
SPSAV   000250R     002 S4.EDT= 010000    T.RCVL  000012    UNIT    000244R   002 U.MEDI= 000040
SYSFDB= ****** GX       S4.HFF= 000020    T.RRFL  000072    US.ABO= 000001        U.MLUN= 000050
S$$WRG= 000000          S4.HFL= 000007    T.SAST  000054    US.BSP= 000002        U.MUP   177772
S$$YSZ= 007600          S4.HHT= 000040    T.SRCT  000071    US.BSY= 000200        U.OWN   177776
S1.DEC= 001000          S4.HSY= 000200    T.STAT  000032    US.FOR= 000040        U.RBNS= 000112
S1.DPR= 000400          S4.RGS= 020000    T.ST2   000034    US.FRK= 000002        U.RCTC= 000113
S1.DSI= 004000          S4.SFC= 040000    T.ST3   000036    US.KPF= 000001        U.RCTS= 000110
S1.ESC= 000002          S4.VFL= 000010    T.TCBL  000030    US.LAB= 000004        U.RED   000002
S1.IBF= 002000          S5.ABP= 100000    T.TIO   000057    US.MDE= 000002        U.RED2  000034
S1.IBY= 000200          S5.BCC= 020000    T.TKSZ  000060    US.MDM= 000020        U.SCB   000020
S1.OBY= 000100          S5.DAO= 040000    T.UCB   000026    US.MNT= 000100        U.SHST= 000076
S1.PTH= 000010          S5.HPC= 000014    T2.ABO= 000100    US.OFL= 000001        U.SHUN= 000074
S1.RES= 010000          S5.HPO= 000020    T2.AST= 100000    US.PUB= 000004        U.SPC = 000036
S1.RNE= 000020          S5.ITI= 000100    T2.CAF= 000400    US.PWF= 000010        U.STS   000005
```

```
		.TITLE	VSET - VNP SETUP CSR AND MICROCODE LOAD BLOCKS
		.IDENT	/V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;	VNP - SETUP CSR AND MICRO-CODE BLOCKS
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00	27-FEB-78
;	VERSION 2.0 RELEASE
;
; 2.00	14-DEC-79
;	DECNET-11M/S V3.0
;	DECNET-11M-PLUS V1.0
;
; 3.00	16-APR-82
;	DECNET-11M V3.1
;	DECNET-11M-PLUS V1.1
;
; 4.00	07-NOV-83
;	DECNET-11M V4.0
;	DECNET-11M-PLUS V2.0
;
; 5.00	22-JUL-85
;	DECnet-11M/S V4.2
;	DECnet-11M-Plus V3.0
;	DECnet-Micro/RSX V1.0
;
;
```

```
    217 000000                                         KEYWD$  ''VECTOR'',             K$VEC
    218 000000                                         KEYWD$  ''VERIFICATION'',       K$VER
    219 000000                                         KEYWD$  ''X25'',                K$X25
    220 000000                                         KEYWD$  ''X29'',                K$X29
    834 000000                          STATE$
    835 000000                           TRAN$   '/,          PAS
    836 000000                           TRAN$   $LAMDA,      DESIR1
    837 000000                          STATE$   PAS
    838 000000                           TRAN$   !DSTR,       ,          $DPAS
    839 000000                          STATE$
    840 000000                           TRAN$   '/,          ACC
    841 000000                           TRAN$   $LAMDA,      DESIR1
    842 000000                          STATE$   ACC
    843 000000                           TRAN$   !DSTR,       ,          $DACC
    844 000000                          STATE$
    845 000000                           TRAN$   '/,          DESIR1
    846 000000                           TRAN$   $LAMDA
    847 000000                          STATE$   DESIR1
    848 000000                           TRAN$   !NODEND,     $EXIT,     BL.AUT
    849 000000                           TRAN$   $LAMDA,      $EXIT,     BL.AUT
    850                            ;
    851                            ; LINE ID
    852                            ;
    853 000000                          STATE$   LINID
    854 000000                           TRAN$   $RAD50,      ,          $LNAME
    855 000000                          STATE$
    856 000000                           TRAN$   <'->
    857 000000                          STATE$
    858 000000                           TRAN$   $DNUMB,      LIN1,      $CTL
    859 000000                           TRAN$   <'*>,        ,              ,      WC.CNT,      $QUAL
    860 000000                          STATE$   LIN1
    861 000000                           TRAN$   <'->,        LIN2
    862 000000                           TRAN$   $LAMDA,      $EXIT,         ,      FL.CNT,      $FLAGS
    863 000000                          STATE$   LIN2
    864 000000                           TRAN$   $DNUMB,      $EXIT,     $UNT
    865 000000                           TRAN$   <'*>,        $EXIT,     $XUNT,     WC.UNT,      $QUAL
    866                            ;
    867                            ; LOGGING ID
    868                            ;
    869 000000                          STATE$   LOGID
    870 000000                           TRAN$   &K$MON,      $EXIT,         ,      OP$MON,      $OPTON
    871 000000                           TRAN$   &K$CNSL,     $EXIT,         ,      OP$CON,      $OPTON
    872 000000                           TRAN$   &K$FIL,      $EXIT,         ,      OP$FIL,      $OPTON
    873                            ;
    874                            ; MODULE ID
    875                            ;
    876 000000                          STATE$   MODID
    877 000000                           TRAN$    &K$X25,     MODID5
    878 000000                           TRAN$    &K$X29,     MODID9,        ,      OP$X9S,      $OPTON
    879
    880 000000                          STATE$   MODID5
    881 000000                           TRAN$    <'->,       ,          STMAT1
    882 000000                          STATE$
    883 000000                           TRAN$    &K$ACES,    $EXIT,     STMAT3,    OP$XAC,      $OPTON
    884 000000                           TRAN$    &K$PRTO,    $EXIT,     STMAT3,    OP$XPR,      $OPTON
    885 000000                           TRAN$    &K$SVR,     $EXIT,     STMAT3,    OP$X5S,      $OPTON
    886
    887 000000                          STATE$   MODID9
    888 000000                           TRAN$    <'->,       MODI91,    STMAT1
    889 000000                           TRAN$     $LAMDA,    $EXIT,     STMAT3
    890 000000                          STATE$   MODI91
```

```
   1513 000000                                   STATE$  EVEID1
   1514 000000                                     TRAN$  $EOS,          EVEID2
   1515 000000                                     TRAN$  $BLANK,        EVEID2
   1516 000000                                     TRAN$  <',     >
   1517 000000                                   STATE$
   1518 000000                                     TRAN$  !EVELIS,       EVEID1
   1519 000000                                   STATE$  EVEID2
   1520 000000                                     TRAN$  $LAMDA,        $EXIT,      BL.AUT
   1521                                  ;
   1522                                  ;       EVENT LIST
   1523                                  ;
   1524 000000                                   STATE$  EVELIS
   1525 000000                                     TRAN$  !EVERAN,       $EXIT
   1526 000000                                     TRAN$  !EVENUM,       $EXIT
   1527                                  ;
   1528                                  ;       EVENT RANGE
   1529                                  ;
   1530 000000                                   STATE$  EVERAN
   1531 000000                                     TRAN$  $DNUMB,        ,           CKEVEB
   1532 000000                                   STATE$
   1533 000000                                     TRAN$  <'->
   1534 000000                                   STATE$
   1535 000000                                     TRAN$  $DNUMB,        $EXIT,      CKEVER
   1536                                  ;
   1537                                  ;       EVENT NUMBER
   1538                                  ;
   1539 000000                                   STATE$  EVENUM
   1540 000000                                     TRAN$  $DNUMB,        $EXIT,      CKEVEN
```

```
                                                                                   G 6
VSTB        CREATED BY  MACRO  ON 29-JUN-85 AT 03:03      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

LO.RET     000064           #5-62
LO.RPA     000030           #5-62
LO.SCR     000016           #5-62
LO.SEG     000054           #5-62
LO.SKA     000014           #5-62
LO.SKN     000014           #5-62
LO.SNM     000035           #5-62
LO.TPA     000020           #5-62
LO.TRB     000034           #5-62
LO.TRI     000030           #5-62
LO.UNT     000033           #5-62
LO.VCT     000014           #5-62
LO.VER     000042           #5-62
LR.ACC     000020           #5-62
LR.ACO     000052           #5-62
LR.ADD     000012           #5-62
LR.ALI     000002           #5-62
LR.BLK     000010           #5-62
LR.CTL     000004           #5-62
LR.DES     000010           #5-62
LR.HLP     000002           #5-62
LR.LIN     000002           #5-62     5-62      5-62      5-62      5-62      5-62      5-62
LR.NAM     000012           #5-62
LR.PAS     000041           #5-62
LR.PRO     000002           #5-62
LR.STS     000000           #5-62
LR.TRI     000006           #5-62
LR.TYP     000002           #5-62
LR.UCB     000016           #5-62
LR.UIC     000020           #5-62
LR.UNT     000005           #5-62
LS.ALI   = 000016           #5-62
LS.CEX   = 000001           #5-62
LS.CIR   = 000033           #5-62
LS.CXO   = 010000           #5-62
LS.CX5   = 000400           #5-62
LS.ECH   = 001000           #5-62
LS.FDX   = 004000           #5-62
LS.HDX   = 010000           #5-62
LS.HLP   = 000012           #5-62
LS.LIN   = 000003           #5-62
LS.LMC   = 000007           #5-62
LS.LOG   = 000020           #5-62
LS.MOD   = 000036           #5-62
LS.NOD   = 000014           #5-62     19-1641
LS.NTI   = 000013           #5-62
LS.OBJ   = 000022           #5-62
LS.OPT   = 000400           #5-62
LS.PRO   = 000002           #5-62
LS.TOP   = 002000           #5-62
LS.TST   = 000010           #5-62
LS.UNF   = 020000           #5-62
```

```
                                                                                   H 6
VSTB        CREATED BY  MACRO  ON 29-JUN-85 AT 03:03      PAGE 3
```

```
     54                                                 .SBTTL  MACRO DEFINITIONS
     55                                         ;***
     56                                         ; LIBRARY MACROS
     57                                         ;***
     58                                                 .MCALL  NHWDF$,PDVDF$,SLTDF$
     59
     60                                         ;***
     61                                         ; GLOBAL LIBRARY SYMBOLS
     62                                         ;***
     63 000000                                          NHWDF$  ,<=>            ; HARDWARE REGISTERS
     64 000000                                          PDVDF$  <=>,<:>,LIST    ; PROCESS DESCRIPTION VECTOR OFFSETS
                                                ;
                                                ;       PROCESS DESCRIPTOR VECTOR OFFSETS AND BIT DEFINITIONS
                                                ;

        000000                                  Z.DSP:: .BLKW   1       ; RELOCATION BIAS OF DISPATCH TABLE
        000002                                          .BLKW   1       ; VIRTUAL ADDRESS OF DISPATCH TABLE

                                                        .IF DF  X$$MCB
                                                Z.QUE:: .BLKW   1       ; ADDRESS OF SCHEDULER LEVEL QUEUE
                                                        .ENDC
        000004                                  Z.NAM:: .BLKW   1       ; PROCESS NAME (RAD50)
        000006                                  Z.LLN:: .BLKB   1       ; # OF LOGICAL LINES (LLC'S ONLY)
        000007                                  Z.SCH:: .BLKB   1       ; PROCESS PRIORITY (COMPLEMENT)
        000010                                  Z.FLG:: .BLKW   1       ; FLAGS WORD
                                                        .IF DF  X$$MCB!R$$11M!R$$11S
        000012                                  Z.PCB:: .BLKW   1       ; PCB OF LOADED PROCESS PARTITION
                                                        .ENDC
                                                        .IF DF  R$$11D!I$$AS
                                                Z.GCD:: .BLKW   1       ; ADDRESS OF GCD NODE( 0 = PROCESS NOT RESIDENT )
                                                        .ENDC
        000014                                  Z.AVL:: .BLKW   1       ; POINTER TO FIRST FREE BLOCK IN PROCESS SPACE
                        000016                  Z.LEN== .-Z.DSP         ; LENGTH OF PDV
        000016                                  Z.DAT:: .BLKW   1       ; BIAS OF LLC DATA BASE
        000020                                  Z.MAP::                 ; DISPLACEMENT OF LLC MAPPING TABLE
                                                ;
                                                ; FLAGS WORD BIT DEFINTIONS
                                                ;
                        000000                  ZF.X3P==   0            ; PROCESS REQUIRES PARAMETERS FROM X3P$DF MACRO
                        000001                  ZF.DDM==   1            ; PROCESS IS A DDM
                        000002                  ZF.DLC==   2            ; PROCESS IS A DLC
                        000004                  ZF.LLC==   4            ; PROCESS IS A LLC
                        000010                  ZF.MFL==  10            ; PROCESS IS MARKED FOR LOAD ( WITH NS )
                        000020                  ZF.KMX==  20            ; PROCESS IS A DDM FOR A KMC DEVICE
                        000040                  ZF.MUX==  40            ; PROCESS IS A DDM FOR A MULTIPLEXER DEVICE
                        000100                  ZF.LMC== 100            ; PROCESS REQUIRES MICRO-CODE TO BE LOADED
                        000200                  ZF.TIM== 200            ; PROCESS REQUESTS TIMER SUPPORT
                        000400                  ZF.MTM== 400            ; PROCESS REQUIRES TIMER ENTRY ON EVERY ACTIVE PROCESSOR
                                                                        ; (RSX-11M+ MULTI-PROCESSOR SYSTEMS ONLY)
                        001000                  ZF.COU==1000            ; LLC/DLC PROCESS SUPPORTS COUNTERS
                        002000                  ZF.PSE==2000            ; PROCESS IS A PSEUDO DDM/DLC
                        004000                  ZF.DIA==4000            ; PROCESS REQUIRES DIAGNOSTICS MICRO-CODE LOAD
                        010000                  ZF.SLI==10000           ; PROCESS SUPPORTS SYSTEM LEVEL INTERFACE
                        020000                  ZF.MAN==20000           ; PROCESS SUPPORTS NETWORK MANAGEMENT REQUESTS
                        040000                  ZF.INI==40000           ; PROCESS SUPPORTS INITIALIZATION AND TERMINATE
                        100000                  ZF.DVP==100000  ; PROCESS ALWAYS RUNS AT IT'S PRIORITY
                                                ;
```

Table of contents

```
     5-   54    MACRO DEFINITIONS
     6-   64    LOCAL DATA
     7-  172    ERROR MESSAGES
     8-  207    $TMP - TEMPLATE FILE PROCESSOR
     9-  283    IMGRES - REWRITE IMAGE
    10-  311    $BUFCK - CHECK NEXT 128. BYTES
    11-  405    TM.RD - READ RECORD
    12-  440    $ELINE - ECHO LINE
    13-  455    $RQ(N) - REQUEST N BYTES
    14-  507    $EVEN - .EVEN ACTION ROUTINE
    15-  527    $VFDDM - VERIFY DDM
    16-  543    $VFDLC - VERIFY DLC
    17-  560    $VFSPC - VERIFY SPECIAL OPERATOR
    17-  561    $VFLV1 - VERIFY LEVEL 1
    18-  593    $TERR - PRINT ERROR MESSAGE
    18-  594    $TWARN - PRINT WARNING
```

```
                                          .SBTTL  $RQ(N) - REQUEST N BYTES
                                  ;+
                                  ; $RQ(N) - REQUEST ACCESS TO BINARY BUFFER
                                  ;
                                  ; THESE ROUTINES ARE CALLED TO REQUEST ACCESS TO THE BINARY BUFFER,
                                  ; WHICH IS LOCATED AT THE END OF THE TASK'S SPACE. THE CALLER SPECIFIES
                                  ; THE NUMBER OF BYTES HE WISHES TO ADD TO THE BUFFER BY USING THE
                                  ; APPROPRIATE ENTRY POINT.
                                  ;
                                  ; INPUTS:
                                  ;       $PTR=CURRENT POINTER INTO BINARY BUFFER
                                  ;       $LEFT=SPACE REMAINING IN BINARY BUFFER
                                  ;       F1.BBO SET IF BUFFER ALREADY OVERFLOWED
                                  ;
                                  ; OUTPUTS:
                                  ;       C-BIT=SUCCESS/FAILURE
                                  ;       R2=POINTER INTO BINARY BUFFER
                                  ;       $PTR INCREMENTED BY CALLER'S REQUESTED SIZE
                                  ;       $LEFT DECREMENTED BY CALLER'S REQUESTED SIZE
                                  ;       F1.BBO, F1.ERR SET IF THIS REQUEST OVERFLOWS BUFFER
                                  ;-

                                          .ENABL  LSB

001170  012746  000007            $RQ7::  MOV     #7,-(SP)
001174  000416                            BR      10$
001176  012746  000005            $RQ5::  MOV     #5,-(SP)
001202  000413                            BR      10$
001204  012746  000004            $RQ4::  MOV     #4,-(SP)
001210  000410                            BR      10$
001212  012746  000003            $RQ3::  MOV     #3,-(SP)
001216  000405                            BR      10$
001220  012746  000002            $RQ2::  MOV     #2,-(SP)
001224  000402                            BR      10$
001226  012746  000001            $RQ1::  MOV     #1,-(SP)
001232  032767  000001  000000'   10$:    BIT     #F1.BBO,$FLAG1    ; HAS BUFFER ALREADY OVERFLOWED?
001240  001011                            BNE     20$               ; IF NE, YES
001242  161667  000010'                   SUB     (SP),$LEFT        ; ENOUGH ROOM FOR THIS REQUEST?
001246  002010                            BGE     30$               ; IF GE, YES
001250                                    EMSG$   13                ; BINARY BUFFER OVERFLOW
001256  052767  000001  000000'           BIS     #F1.BBO,$FLAG1    ; INDICATE OVERFLOW
001264  005426                    20$:    NEG     (SP)+             ; PURGE STACK (SETS C-BIT)
001266  000410                            BR      40$
001270  005767  000000'           30$:    TST     $FLAG1            ; HAVE ANY OTHER ERRORS OCCURRED?
001274  100773                            BMI     20$               ; IF MI, YES
001276  016702  000006'                   MOV     $PTR,R2           ; RETURN CURRENT POINTER VALUE
001302  062667  000006'                   ADD     (SP)+,$PTR        ; ADJUST $PTR UP
001306  105012                            CLRB    (R2)              ; STORE EOF INDICATOR (CLEARS C-BIT)
001310                            40$:    RETURN

                                          .DSABL  LSB
```

Table of contents

```
     5-   54    MACRO DEFINTIONS
     6-   66    LOCAL DATA
     7-   80    ERROR MESSAGES
     8-  104    $TM.IN - INITIALIZE TEMPLATE DATA
     9-  193    FNDADD - FIND LINE TABLE
```

Table of contents

6- 172 $VTST - VNP TESTING ROUTINES

```
                                                                     G 12
VTST      CREATED BY  MACRO  ON 29-JUN-85 AT 03:09     PAGE 2
SYMBOL CROSS REFERENCE                                 CREF    04.00

SYMBOL   VALUE             REFERENCES

                            6-339    6-340    6-345    6-346    6-351    6-353    6-362    6-367    6-368
                            6-370    6-382    6-392    6-396    6-397    6-398    6-399    6-409    6-417
                            6-421    6-422    6-423    6-424    6-434
NEWLIN     002042 R         6-319   #6-325    6-351
NO       = 000000          #5-70     6-313    6-314    6-319    6-345    6-346    6-351
NUL        002052 R         6-321   #6-328    6-353
N.FID    = ******   GX      6-439
OB       = 015010          #5-71     6-313
ODT        000764 R         6-251   #6-257
OW       = 031010          #5-73     6-306    6-338    6-345    6-367    6-370    6-397    6-422
PROMPT     004026 R         6-184    6-267    6-268    6-274    6-275    6-281    6-282    6-283    6-289
                            6-290    6-291    6-297    6-298    6-372    6-389    6-404    6-405    6-414
                            6-429    6-430   #6-465
PUTAD      001326 R         6-235   #6-289
PUTRC      001436 R         6-233   #6-297
Q.IOFN   = 000002           5-59
Q.IOPL   = 000014           5-60     5-61     5-62
SSTTBL     000030 R        #5-167    6-257
SYSFDB   = ******   GX      6-439   *6-446   *6-447   *6-448   *6-450   *6-452   *6-454
TAB        002044 R        #6-326
TST        000036 R        #6-184    6-204    6-224
VECTOR     000674 R         6-197   #6-226
VF.INI   = ******   GX      6-441
VF.VDT   = ******   GX      6-186    6-190    6-438
X10        002552 R        #6-370    6-384
X101       001132 R         6-267    6-268   #6-270
X102       001214 R         6-274    6-275   #6-277
X103       001324 R         6-281    6-282    6-283   #6-285
X104       001434 R         6-289    6-290    6-291   #6-293
X105       001516 R         6-297    6-298   #6-300
X106       002772 R         6-391   #6-394
X107       003152 R         6-389    6-393   #6-400
X108       003262 R         6-404    6-405    6-408   #6-410
X109       003346 R         6-416   #6-419
X110       003526 R         6-414    6-418   #6-425
X111       003636 R         6-429    6-430    6-433   #6-435
X20        002652 R         6-379   #6-382
X30        002702 R         6-376    6-381   #6-383
X50        002706 R         6-373   #6-385
$ALL16   = ******   GX      6-390
$ALL22   = ******   GX      6-415
$BFR     = ******   GX      6-311    6-343    6-369    6-454
$CBTA    = ******   GX      6-483
$CLBUF   = ******   GX      6-198    6-470   *6-474    6-482    6-484
$CLIOS   = ******   GX      6-374
$CLPMT   = ******   GX      6-466   *6-467   *6-476
$COTB    = ******   GX      6-267    6-268    6-274    6-275    6-281    6-282    6-283    6-289    6-290
                            6-291    6-297    6-298    6-377    6-389    6-404    6-405    6-414    6-429
                            6-430
$CRAVL   = ******   GX      6-452
$DEA16   = ******   GX      6-407
$DEA22   = ******   GX      6-432
```

```
                                                                     H 12
VTST      CREATED BY  MACRO  ON 29-JUN-85 AT 03:09     PAGE 3
```

```
VUMR        CREATED BY  MACRO  ON 29-JUN-85 AT 03:09     PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE               REFERENCES

F.NRBD  = ******  GX        *6-135     *6-140     *9-251     *9-264
F.RSIZ  = ******  GX        *10-292
F.URBD  = ******  GX        *6-139     *9-263     *10-291    *10-294
GFIRST    000304 R           6-105      9-241     #10-277
GNEXT     000354 R           6-108      9-243     #10-300
M.LGTH  = 000014             5-74       5-75       10-292     10-306
M.UMRA    000002             6-110      6-115     *6-119
M.UMRN    000004             6-111      6-117
M.UMVH    000010            *6-134
M.UMVL    000006            *6-133
NEXT      000014 R          #5-75       6-115      9-250      10-294     10-304
PREV      000000 R          #5-74       6-110      6-111     *6-120     *6-121      6-135      9-245     *9-250      9-251
                             10-291     10-305     10-314
R$$MPL  = ******             7-159      9-253      10-277
SYSFDB  = ******  GX        *6-135     *6-139     *6-140     *9-251     *9-263     *9-264     *10-291    *10-292    *10-294
UBMPR   = 170200            #5-80       6-112      6-126
$ASUMR    000000 RG         #6-104
$BFR    = ******  GX         6-139      6-140      9-263      9-264
$DEUMR    000172 RG         #9-238
$GETRV  = ******  GX         10-292     10-318
$PUTRC  = ******  GX         6-137      9-261
$RADDR  = ******  GX        *6-137     *9-261     *10-292    *10-318
$SAVAL  = ******  GX         6-104      9-238
.UMRHD  = ******  GX         10-287
```

```
54
55                          .SBTTL  MACRO DEFINITIONS
56                  ;***
57                  ; LIBRARY MACROS
58                  ;***
59                          .MCALL  ASR$,GETRV,PUTRC,GETAD,PUTAD,SWTCHO,SWTCHI,SCBDF$,CEACC$
60                          .MCALL  ERMSG$,ERRPT$,ERBLK$,OBJDF$,RNBDF$,ANBDF$,FLTDF$,CALLR
61                          .MCALL  ECDDB$,XPDDB$,GETRC,TCBDF$,PLBDF$,CTRDF$,DHBDF$,PCBDF$
62                          .MCALL  ADJDF$,VECDF$,RESRG,SAVRG
63                          .ENABL  LC
64
65                          .IF DF  R$$MPL                                                  ;[MP01]
66                          .MCALL  KRBDF$,CTBDF$                                           ;[MP01]
67                          KRBDF$                  ; DEFINE KRB OFFSETS                    ;[MP01]
68                          CTBDF$                  ; DEFINE CTB OFFSETS
69                          .IFF                                                            ;[MP01]
70                          .MCALL  NKRDF$                                                  ;[M[01]
71 000000                   NKRDF$                                                          ;[MP01]
72
73                          .ENDC                                                           ;[MP01]
74
75
76 000000                   SCBDF$                  ; STATUS CONTROL BLOCK OFFSETS
77 000000                   ADJDF$                  ; DEFINE ADJACECY DATA BASE
78 000000                   TCBDF$                  ; DEFINE TCB OFFSETS
79 000000                   OBJDF$                  ; DEFINE OBJECT BLOCK OFFSETS
80 000000                   FLTDF$                  ; DEFINE FILTER BLOCK OFFSETS
81 000000                   ANBDF$                  ; DEFINE ALIAS BLOCK OFFSETS
82 000000                   RNBDF$                  ; DEFINE REMOTE BLOCK OFFSETS
83 000000                   ECDDB$                  ; DEFINE ECL DDB OFFSETS
84 000000                   PLBDF$                  ; DEFINE PLB OFFSETS
85 000000                   CTRDF$                  ; ECL NODE COUNTER BLOCK OFFSETS
86 000000                   XPDDB$                  ; DEFINE XPT DDB OFFSETS
87 000000                   DHBDF$                  ; DEFINE DECNET HOME BLOCK OFFSETS
88 000000                   PCBDF$                  ; DEFINE PCB OFFSETS
89 000000                   VECDF$                  ; DEFINE VECTOR BLOCK OFFSETS
90
```

```
                                              .SBTTL  $CPRO - CLEAR PROCESS
                                      ;+
                                      ; $CPRO - "CLEAR PROCESS"
                                      ;
                                      ; THIS ROUTINE IS CALLED TO REMOVE A PROCESS FROM CORE. THE MARK-FOR-LOAD
                                      ; BIT IN THE PDV IS ALWAYS CLEARED.
                                      ;
                                      ; INPUTS:
                                      ;        $PDVA=PDV ADDRESS
                                      ;
                                      ; OUTPUTS:
                                      ;        NONE
                                      ;-
001462                                $CPRO::
001462  026727  000000G 004640                CMP     $NAME,#^RAUX     ; YOU CANNOT CLEAR AUX
001470  001411                                BEQ     101$             ; ...
001472  016704  000000G                       MOV     $PDVA,R4         ; GET PDV ADDRESS
001476                                        CALL    XPRO             ; RELEASE PROCESS FROM MEMORY
001502  142764  000000G 000000G               BICB    #ZF.MFL,Z.FLG(R4) ; CLEAR THE MARK-FOR-LOAD FLAG
001510  000241                                CLC                      ; RETURN WITH C-BIT CLEAR
001512                                99$:    RETURN

                                      ; ERRORS

001514  012746  001512'               101$:   MOV     #99$,-(SP)       ; SET RETURN ADDRESS
001520                                        ERRPT$  ERRA,$EROUT      ; WRITE MESSAGE
001530                                        ERMSG$  <The AUX process cannot be cleared explicitly.>
001605                                        ERBLK$  ERRA
```

```
                                                CMP     (R4)+,#13746     ; OUR ICB ?                           ;[MP01]
                                                BNE     20$              ; IF NO - BRANCH                      ;[MP01]
                                                CMP     (R4)+,#172352    ; OUR ICB ?                           ;[MP01]
                                                BNE     20$              ; IF NO - BRANCH                      ;[MP01]
                                                CMP     (R4)+,#12737     ; OUR ICB ?                           ;[MP01]
                                                BNE     20$              ; IF NO - BRANCH                      ;[MP01]
                                                MOV     ..PDV,-(SP)      ; GET PDV ADDRESS                     ;[MP01]
                                                ADD     #Z.DSP,(SP)      ; POINT TO DISPATCH                   ;[MP01]
                                                CMP     (R4)+,@(SP)+     ; OUR ICB ?                           ;[MP01]
                                                BNE     20$              ; IF NO - BRANCH                      ;[MP01]
                                                CMP     (R4)+,(R4)+      ; POINT TO LINE TABLE JMP INSTRUCTION ;[MP01]
                                                CMP     (R4)+,(R4)+      ;  ...                                ;[MP01]
                                                BR      33$              ; CONTINUE                            ;[MP01]

                                                .ENDC                                                          ;[MP01]


 003254  022427  172352                 35$:    CMP     (R4)+,#172352    ; KISAR5?
 003260  001344                                 BNE     20$              ; NO IF NE
 003262  022427  012737                         CMP     (R4)+,#12737     ; IS IT 'MOV'?
 003266  001341                                 BNE     20$              ; NO IF NE
 003270  016746  000202'                        MOV     ..PDV,-(SP)      ; PROCESS' PDV
 003274  062716  000000G                        ADD     #Z.DSP,(SP)      ; POINT AT BIAS
 003300  022436                                 CMP     (R4)+,@(SP)+     ; DO BIAS'S MATCH?
 003302  001333                                 BNE     20$              ; NO IF NE
 003304  022427  172352                         CMP     (R4)+,#172352    ; KISAR5?
 003310  001330                                 BNE     20$              ; NOPE IF NE
 003312  022427  000137                 33$:    CMP     (R4)+,#137       ; 'JMP'
 003316  001325                                 BNE     20$              ; NOPE IF NE
 003320  021402                                 CMP     (R4),R2          ; (R4) NOW POINTING AT LINE TABLE
 003322  103723                                 BLO     20$              ; NOT OURS, IF LO
 003324  021403                                 CMP     (R4),R3          ; WITHIN RANGE?
 003326  103321                                 BHIS    20$              ; NO IF HIS
 003330  066767  000000G 000176'                ADD     $ICBLN,ICBSZ     ; INCREASE SIZE OF THIS ICB
 003336  000241                         36$:    CLC                      ; C-CLEAR THEN FOUND ONE
 003340  000401                                 BR      50$
 003342  000261                         40$:    SEC                      ; C-SET THEN NO MATCH
 003344                                 50$:    SWTCHI                   ; REVERT TO #$3UF
 003352                                         RETURN

```

```
                                                                     H  1
VRQB - VNP PROCESS REQUEST BLOC MACRO V05.03b Tuesday 30-Jul-85 10:53  Page 35-3
Symbol table

U.ST2   000007          U2.L3S= 000004          $C5TA = ****** GX           $FPDV   002776R         $$PAR = ****** GX
U.SUB = 000036          U2.L8S= 010000          $DDMPV= ****** GX           $FSLT   003054RG        $$PRI = ****** GX
U.TCHP  000042          U2.NEC= 004000          $DEA16= ****** GX           $FUNC = ****** GX       $$SNUM= ****** GX
U.TCVP  000043          U2.PRV= 000010          $DEVHD= ****** GX           $GETRV= ****** GX       $$TBL = ****** GX
U.TFLK  000040          U2.RMT= 020000          $DLCPV= ****** GX           $LLCTA= ****** GX       $$VECT= ****** GX
U.TFRQ  000036          U2.R04= 100000          $ERR01  002036R         002 $MISS = ****** GX       $$$   = 000062
U.TIXL  000060          U2.SCS= 000004          $ERR02  002074R         002 $MXVCT= ****** GX       .CALI   000224R     002
U.TLPP  000053          U2.SLV= 000200          $ERR03  002134R         002 $NAME = ****** GX       .CCEX   000132R     002
U.TMTI  000047          U2.VT5= 000002          $ERR04  002210R         002 $NLRQB  000000RG        .CLIN   000146R     002
U.TRCK= 000100          U2.7CH= 010000          $ERR05  002252R         002 $OPTON= ****** GX       .CLOG   000170R     002
U.TSTA  000026          U3.OPA= 100000          $ERR06  002320R         002 $PDVA = ****** GX       .CNOD   000162R     002
U.TST5  000054          U3.PAR= 040000          $ERRZB  001700R         002 $PDVNM= ****** GX       .COBJ   000224R     002
U.TST6  000056          U3.UPC= 020000          $ERRZP  001734R         002 $PDVTA= ****** GX       .CPRO   000136R     002
U.TTAB  000050          U4.CR = 000100          $ERRZR  001772R         002 $PSIPT= ****** GX       .DALI   000224R     002
U.TTYP  000046          VF.ADD= ****** GX       $ERRZ1  001410R         002 $PUTRC= ****** GX       .DCEX   000224R     002
U.TUX   000024          VF.PEM= ****** GX       $ERRZ3  001440R         002 $RADDR= ****** GX       .DCIR   000224R     002
U.UHVR= 000107          V$$CTR= 001000          $ERRZ4  001466R         002 $RERR   003250RG        .DLIN   000224R     002
U.UIC   000044          WC.CNT= 000400          $ERRZ5  001530R         002 $RSIZE= ****** GX       .DLOG   000224R     002
U.UNFL= 000052          WC.EVE= 004000          $ERRZ6  001550R         002 $SAVAL= ****** GX       .DMOD   000176R     002
U.UNIT  000006          WC.TRI= 002000          $ERRZ7  001604R         002 $SAVRG= ****** GX       .DNOD   000224R     002
U.UNSZ= 000114          WC.UNT= 001000          $ERRZ8  001644R         002 $SAVVR= ****** GX       .DOBJ   000224R     002
U.UNTI= 000060          X$$DBT= 000000          $ERR86  001350R         002 $SLN  = ****** GX       .DPRO   000224R     002
U.USVR= 000106          ZF.DDM= ****** GX       $ERR87  001302R         002 $SLTA = ****** GX       .NOP    000156R     002
U.UTIL= 000036          ZF.DLC= ****** GX       $ERR88  001220R         002 $SLTMA= ****** GX       .RNTI   000230R     002
U.VCB = 000034          ZF.KMX= ****** GX       $ERR90  001164R         002 $SLTNM= ****** GX       .SALI   000224R     002
U.VSER= 000120          ZF.LMC= ****** GX       $ERR91  001126R         002 $TIOUT= ****** GX       .SCEX   000076R     002
U.2MED= 000070          ZF.MUX= ****** GX       $ERR92  001064R         002 $TRIB = ****** GX       .SCIR   000204R     002
U2.AT.= 000020          Z.DAT = ****** GX       $ERR93  001020R         002 $VFLAG= ****** GX       .SLIN   000116R     002
U2.CRT= 002000          Z.FLG = ****** GX       $ERR94  000764R         002 $$CSR = ****** GX       .SLOG   000170R     002
U2.DH1= 100000          Z.LLN = ****** GX       $ERR95  000730R         002 $$DEV = ****** GX       .SMOD   000216R     002
U2.DT1= 040000          Z.MAP = ****** GX       $ERR96  000652R         002 $$EXT = ****** GX       .SNOD   000162R     002
U2.DZ1= 000100          Z.NAM = ****** GX       $ERR97  000614R         002 $$INC = ****** GX       .SOBJ   000224R     002
U2.ESC= 001000          Z.PCB = ****** GX       $ERR98  000540R         002 $$LINX= ****** GX       .SPRO   000104R     002
U2.HFF= 010000          $$BFR = ****** GX       $ERR99  000506R         002 $$LRB = ****** GX       .TLGTH= ****** GX
U2.HLD= 000040          $BITS   002744R         $FLAGS= ****** GX           $$MTP = ****** GX       .TSKHD= ****** GX
U2.LOG= 000400          $CBOMG= ****** GX       $FMASK= ****** GX           $$MUX = ****** GX       ..SCSR= ****** GX
U2.LWC= 000001          $CMAND= ****** GX

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        003736    001   (RW,I,LCL,REL,CON)
DATA    002376    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 26
Work  file writes: 32
Size of work file: 17918 Words  ( 70 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:02:55.35
DB2:VRQB.T3,[132,134]VRQB/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VRQB
```

```
                                                                  I  1
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55       PAGE 1
```

```
                                        ;+
                                                $CSRLD  - LOAD CSR ADDRESS OF THIS SYSTEM LINE INTO CSR BLOCK
                                        ;
                                        ; INPUTS:
                                                $$LTA   - SYSTEM LINE TABLE ADDRESS
                                                $$CSR   - CSR ADDRESS
                                        ;
                                        ; OUTPUTS:
                                                BLOCK POINTED TO BY NTL HOME BLOCK + .CXCSR IS UPDATED
                                        ;
                                        ; NOTE:
                                                ONE REQUIREMENT IS THAT DUPLICATE SLT ENTRIES NOT BE ENTERED
                                                INTO THIS LIST. THE LIST IS MAINTAINED WITH ZERO HOLES ALLOWED
                                                ANYWHERE WITHIN THE LIST. THE FIRST HOLE IS ALWAYS FILLED IN
                                                WITH THE CURRENT CSR BLOCKS BEING ENTERED, BUT THEN THE REST OF
                                                THE LIST IS SCANNED FOR DUPLICATE ENTRIES WHICH ARE REMOVED.
                                        ;-

                                                .MCALL  SWTCHO,PUTAD,SWTCHI

000000                                  $CSRLD::
000000  005005                                  CLR     R5                      ; CLEAR FLAG FOR SEGMENT ALREADY LOADED
000002  012700  000000C                         MOV     #$NTLHB+.CXCSR,R0       ; POINT AT LISTHEAD
000006  012701  000004                          MOV     #4,R1                   ; SET SIZE OF DESIRED SEGMENT
000012  011002                                  MOV     (R0),R2                 ; DOES FIRST BLOCK EXIST ?
000014  001403                                  BEQ     20$                     ; NO
000016  012702  000002G                         MOV     #$UNBLK+2,R2            ; SET UP INITIAL STARTING ADDRESS
000022  160102                                  SUB     R1,R2                   ; ...

000024                                  20$:    CALL    $VUNB                   ; GET NEXT SEGMENT ADDRESS
000030  103003                                  BCC     30$                     ; GOT IT
000032  005705                                  TST     R5                      ; END OF CHAIN, HAS SEGMENT BEEN ADDED ?
000034  001037                                  BNE     90$                     ; YES, 10-4
000036  000772                                  BR      20$                     ; R2 = 0, GO ALLOCATE NEW BLOCK

000040  021267  000000G                 30$:    CMP     (R2),$SLTA              ; IS THIS A DUPLICATE ?
000044  001425                                  BEQ     80$                     ; YES, THEN CHECK FOR DUPLICATE

000046  005705                          40$:    TST     R5                      ; HAS SEGMENT ALREADY BEEN ADDED TO LIST
000050  001365                                  BNE     20$                     ; YES
000052  005712                                  TST     (R2)                    ; IS THIS SEGMENT FREE ?
000054  001363                                  BNE     20$                     ; NO, THEN LOOP

000056  016712  000000G                         MOV     $SLTA,(R2)              ; LOAD SLT ADDRESS
000062  066712  000000G                         ADD     $CETDF,(R2)             ;  ALLOW FOR DIFFERENTIAL
000066  016762  000000G 000002                  MOV     $$CSR,2(R2)             ;  AND CSR ADDRESS
000074  010705                                  MOV     PC,R5                   ; SIGNAL SEGMENT STORED

000076                                  50$:    SWTCHO  #$UNBLK                 ; POINT AT BLOCK BUFFER
000104                                          PUTAD   ,#64.                   ; WRITE UPDATED BLOCK TO SYSTEM IMAGE
000116  000742                                  BR      20$                     ; AND LOOP

000120  005705                          80$:    TST     R5                      ; HAS SEGMENT ALREADY BEEN STORED ?
000122  001407                                  BEQ     90$                     ; NO, THEN WE WERE ALREADY IN LIST
000124  005012                                  CLR     (R2)                    ; REMOVE DUPLICATE ENTRY
000126  005062  000002                          CLR     2(R2)                   ; ...
000132  000761                                  BR      50$                     ; AND CONTINUE
```

```
                                        .SBTTL  COMMANDS

                                        ;       Keyword         Next        Action    Mask (BIS)        Flag to
                                        ;       Type, or        State      Routine      Value             Mask
                                        ;       Substate

                                 ;
                                 ; COMMANDS
                                 ;
000000                                  STATE$  ,&P$VNP
000000                                   TRAN$  &K$CLE,         CLE,         ,          CM$CLE,          $CMAND
000000                                   TRAN$  &K$EXI,         DONE,        ,          LS.XIT,          RQBST
000000                                   TRAN$  &K$HEL,         $EXIT,     $HLPAC,      LS.HLP,          RQBST
000000                                   TRAN$  &K$SET,         SET,         ,          CM$SET,          $CMAND
000000                                   TRAN$  &K$SHO,         SHO,         ,          CM$SHO,          $CMAND
000000                                   TRAN$  &K$VDT,         DONE,        ,          LS.TST,          RQBST
                                 ;
                                 ; UNIVERSAL DONE
                                 ;
000000                                  STATE$  DONE
000000                                   TRAN$  $EOS,           $EXIT
```

```
    891 000000                              TRAN$   &K$SVR,         $EXIT,          STMAT3
    892 000000                              TRAN$   $LAMDA,         $EXIT,          STMAT3
    893                         ;
    894                         ;       NODE ADDRESS
    895                         ;
    896 000000                          STATE$  NODADD
    897 000000                              TRAN$   $DNUMB,         ,               $ARAD1
    898 000000                          STATE$
    899 000000                              TRAN$   <'.>,           NODD
    900 000000                              TRAN$   $LAMDA,
    901
    902 000000                          STATE$  NODX
    903 000000                              TRAN$   $EOS,           $EXIT
    904 000000                              TRAN$   $LAMDA,         $EXIT,          DSTR.E
    905 000000                              TRAN$   $LAMDA,         $EXIT,          STR.B
    906
    907 000000                          STATE$  NODD
    908 000000                              TRAN$   $DNUMB,         NODX,           $ARAD2
    909
    910
    911                         ;
    912                         ;       NODE ID
    913                         ;
    914 000000                          STATE$  NODID
    915 000000                              TRAN$   $LAMDA,         ,               BL.MAN
    916 000000                          STATE$
    917 000000                              TRAN$   !NODADD,        NODID1,         $NADD,          VF.ADD,         $VFLAG
    918 000000                              TRAN$   !NODNAM
    919 000000                          STATE$  NODID1
    920 000000                              TRAN$   <'/>,           NODID3
    921 000000                              TRAN$   !NODEND,        NODID2
    922 000000                              TRAN$   $LAMDA
    923 000000                          STATE$  NODID2
    924 000000                              TRAN$   $LAMDA,         $EXIT,          BL.AUT
    925 000000                          STATE$  NODID3
    926 000000                              TRAN$   $LAMDA,         ,               REJECT
    927                         ;
    928                         ;       NODE ID, DESTINATION VARIANT
    929                         ;
    930                         ;               NOTE:  LEAVES BLANKS ON MANUAL PARSE
    931                         ;
    932 000000                          STATE$  DNODID
    933 000000                              TRAN$   $LAMDA,         ,               BL.MAN
    934 000000                          STATE$
    935 000000                              TRAN$   !NODADD,        DNODI1,         $STADD,         VF.ADD,         $VFLAG
    936 000000                              TRAN$   $RAD50,         DNODI1,         $STDES
    937 000000                          STATE$  DNODI1
    938 000000                              TRAN$   $LAMDA,         $EXIT,          DSTR.E
    939 000000                              TRAN$   $LAMDA,         $EXIT,          STR.B
    940                         ;
    941                         ;       NODE NAME
    942                         ;
    943 000000                          STATE$  NODNAM
    944 000000                              TRAN$   $RAD50,         $EXIT,          $NNAM
    945                         ;
    946                         ;       OBJECT
    947                         ;
```

```
  1542                                  .SBTTL  UTILITY SUBEXPRESSIONS
  1543
  1544                                  ;       Keyword         Next        Action     Mask (BIS)       Flag to
  1545                                  ;       Type, or        State       Routine     Value            Mask
  1546                                  ;       Substate
  1547
  1548                          ;
  1549                          ; UTILITY SUBEXPRESSIONS
  1550                          ;
  1551                          ; DESTINATION NODE STRING, QUOTED OR ANY CHARACTERS OTHER THAN BLANK OR TAB.
  1552                          ; DELIMITED BY SLASHES OR SHORT TERMINATED WITH A BLANK OR TAB OR END OF INPUT
  1553                          ;
  1554                          ;    ASSUMES THAT BLANK PARSING IS ON MANUAL
  1555                          ;
  1556 000000                           STATE$  DSTR
  1557 000000                             TRAN$  $EOS,           $EXIT
  1558 000000                             TRAN$  $LAMDA,         $EXIT,      STR.B
  1559 000000                             TRAN$  $LAMDA,         $EXIT,      DSTR.E
  1560 000000                             TRAN$  $ANY,           DSTR
  1561                          ;
  1562                          ; NODE END CHARACTERS
  1563                          ;
  1564 000000                           STATE$  NODEND
  1565 000000                             TRAN$  <'_>,           $EXIT
  1566 000000                             TRAN$  <':>
  1567 000000                           STATE$
  1568 000000                             TRAN$  <':>,           $EXIT
  1569                          ;
  1570                          ; STRING OR ANY CHARACTERS OTHER THAN BLANK OR TAB
  1571                          ;
  1572 000000                           STATE$  STR
  1573 000000                             TRAN$  $LAMDA,         ,           BL.MAN
  1574 000000                           STATE$
  1575 000000                             TRAN$  !STR.M
  1576 000000                           STATE$
  1577 000000                             TRAN$  $LAMDA,         $EXIT,      BL.AUT
  1578                          ;
  1579 000000                           STATE$  STR.M
  1580 000000                             TRAN$  $EOS,           $EXIT
  1581 000000                             TRAN$  $LAMDA,         $EXIT,      STR.B
  1582 000000                             TRAN$  $ANY,           STR.M
```

```
VSTB      CREATED BY  MACRO  ON 29-JUN-85 AT 03:03     PAGE 3
SYMBOL CROSS REFERENCE                                 CREF   04.00

SYMBOL   VALUE             REFERENCES

LS.XIT  = 000011           #5-62
LX.ALI  = 000017           #5-62
LX.CEX  = 000004           #5-62
LX.CIR  = 000034           #5-62
LX.LIN  = 000006           #5-62
LX.LOG  = 000021           #5-62
LX.NOD  = 000015           #5-62
LX.OBJ  = 000023           #5-62
LX.PRO  = 000005           #5-62
OF$INS  = 000001           #5-62
OF$LOG  = 000100           #5-62
OF$OFF  = 000002           #5-62
OF$ON   = 000000           #5-62
OF$SMC  = 000200           #5-62
OP$ADD  = 000200           #5-62
OP$ALL  = 000200           #5-62
OP$CCS  = 000004           #5-62
OP$CIE  = 004000           #5-62
OP$CON  = 000400           #5-62
OP$COP  = 000001           #5-62
OP$COS  = 000100           #5-62
OP$DST  = 000200           #5-62
OP$DTE  = 000040           #5-62
OP$DUP  = 000002           #5-62
OP$EST  = 000100           #5-62
OP$EVE  = 010000           #5-62
OP$FIL  = 000200           #5-62
OP$GRP  = 000100           #5-62
OP$HOS  = 000001           #5-62
OP$HTM  = 000002           #5-62
OP$INA  = 004000           #5-62
OP$INC  = 001000           #5-62
OP$KDS  = 002000           #5-62
OP$KDT  = 000400           #5-62
OP$KEV  = 001000           #5-62
OP$KGR  = 001000           #5-62
OP$KNT  = 010000           #5-62
OP$KSK  = 002000           #5-62
OP$LCT  = 000200           #5-62
OP$LNE  = 000040           #5-62
OP$LOC  = 000001           #5-62        #5-62
OP$LST  = 000002           #5-62
OP$LTM  = 000001           #5-62
OP$LVL  = 000010           #5-62
OP$MAC  = 000010           #5-62
OP$MCO  = 000002           #5-62
OP$MDE  = 000400           #5-62
OP$MLI  = 000004           #5-62
OP$MON  = 000100           #5-62
OP$MST  = 000020           #5-62
OP$NAM  = 000004           #5-62        #5-62        #5-62
OP$NET  = 004000           #5-62
```

```
                                        ; CHANNEL TABLE FLAGS
                                        ;
                    100000              ZS.ASN==100000  ; CHANNEL IS FREE TO ASSIGN
                    140000              ZS.BSY==140000  ; CHANNEL IS IN PROCESS OF ASSIGNMENT

     65 000000                                  SLTDF$  <=>,<:>,LIST    ; SYSTEM LINE TABLE OFFSETS

                                        ;
                                                SYSTEM LINE TABLE OFFSETS AND FLAG WORD BIT DEFINITIONS
                                        ;
        177776                          L.CVA:: .BLKW   1       ; DDM CHARACTERISTICS VIRTUAL ADDRESS
        000000                          L.FLG:: .BLKW   1       ; FLAGS WORD
        000002                          L.DDM:: .BLKB   1       ; DDM PROCESS INDEX
        000003                          L.DLC:: .BLKB   1       ; DLC PROCESS INDEX
        000004                          L.DDS:: .BLKW   1       ; DDM LINE TABLE ADDRESS
        000006                          L.DLM:: .BLKW   1       ; DLC LINE TABLE RELOCATION BIAS
        000010                          L.DLS:: .BLKW   1       ; DLC LINE TABLE ( VIRTUAL ) ADDRESS
        000012                          L.CTL:: .BLKB   1       ; CONTROLLER NUMBER
        000013                          L.UNT:: .BLKB   1       ; MULTIPLEXER UNIT NUMBER
        000014                          L.NSTA::        .BLKB   1       ; NUMBER OF TRIBUTARIES ON THIS LINE
        000015                          L.COST::        .BLKB   1       ; LINE COST FOR ROUTING TASK
        000016                          L.KRBA::        .BLKW   1       ; POINTER TO CONTROLLER REQUEST BLOCK
        000020                          L.NMST::        .BLKB   1       ; NETWORK MANAGEMENT LINE STATE/SUBSTATE
        000021                          L.OWNR::        .BLKB   1       ; LINE OWNER PROCESS INDEX
        000022                          L.MPF::                 ; START OF MULTIPOINT FLAG TABLE
                    000022              L.LEN==.                        ; LENGTH OF EACH SLT ENTRY
                                        ;
                                                FLAG WORD DEFINITIONS
                                        ;
                    000000              LF.X2P==000000  ; LINE NEEDS PARAMETERS FROM X2P$DF
                    000007              LF.BWT==000007  ; BUFFER WAIT QUEUE COUNT
                    000010              LF.TIM==000010  ; LINE NEEDS TIMER SERVICE
                    000020              LF.MTP==000020  ; LINE IS MULTI-POINT
                    000040              LF.SER==000040  ; CIRCUIT HAS SERVICE DISABLED
                    000100              LF.MDC==000100  ; LINE NEEDS MODEM CONTROL
                    000200              LF.PAC==000200  ; LINE WAS PREVIOUSLY ACTIVE
                                                        ; (BEFORE RECONFIGURATION)
                    000400              LF.BRO==000400  ; LINE IS A BROADCAST CHANNEL (ETHERNET)
                    001000              LF.LPB==001000  ; LINE IS IN LOOPBACK
                    002000              LF.ENA==002000  ; LINE IS TO BE ENABLED AT INITIALIZATION TIME
                    004000              LF.MFL==004000  ; LINE IS MARKED FOR LOAD AT INITIALIZATION TIME
                    010000              LF.REA==010000  ; LINE IS REASSIGNABLE
                    020000              LF.UNL==020000  ; LINE IS MARKED FOR UNLOAD
                    040000              LF.RDY==040000  ; LINE IS READY (DDM,DLC PROCESSES AND LINE TABLES ASSIGNED)
                    100000              LF.ACT==100000  ; LINE IS ACTIVE ( LINE IS ASSIGNED TO AN LLC )
                                        ;
                                        ;       NETWORK MANAGEMENT STATE/SUBSTATE BYTE DEFINITIONS
                                        ;           STATE = LOW 4 BITS
                                        ;           SUBSTATE = HIGH 4 BITS
                                        ;
                    000017              LN.STA==17      ; STATE FIELD BIT MASK
                    000360              LN.SUB==360     ; SUBSTATE FIELD BIT MASK
                    000000              LN.ON==0                ; STATE = ON
                    000001              LN.OFF==1       ; STATE = OFF
                    000002              LN.SER==2       ; STATE = SERVICE
                    000003              LN.OAU==3       ; STATE = ON, AUTOSERVICE
                    000004              LN.OOP==4       ; STATE = ON, OPEN
                    000000              LN.CLO==0       ; SUBSTATE = CLOSED
```

```
                .TITLE  VTMP - VNP TEMPLATE PROCESSING
                .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       VNP - TEMPLATE PROCESSING MAIN CONTROL LOOP
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
                                        .SBTTL  $EVEN - .EVEN ACTION ROUTINE
                                ;+
                                ; $EVEN - ''.EVEN'' ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       NONE
                                ;
                                ; OUTPUTS:
                                ;       R2=DESTROYED
                                ;-
001312                          $EVEN::
001312  032767  000001  000012'         BIT     #1,$SIZE        ; IS SIZE CURRENTLY EVEN?
001320  001410                          BEQ     10$             ; IF EQ, YES - DO NOTHING
001322                                  CALL    $RQ2            ; REQUEST BINARY FILE
001326  103405                          BCS     10$             ; IF CS, NO ACCESS
001330  112722  000000G                 MOVB    #.BYTE,(R2)+    ; STORE FUNCTION CODE
001334  105012                          CLRB    (R2)            ; STORE ZERO VALUE
001336  005267  000012'                 INC     $SIZE           ; ADJUST DATA BASE SIZE
001342                          10$:    RETURN
```

```
        .TITLE  VTMPIN - VNP TEMPLATE INITIALIZATION
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       VNP - TEMPLATE PROCESSING INITIALIZATION ROUTINES
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
        .TITLE  VTST - VNP TESTING ROUTINES
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       VNP - SYSTEM IMAGE ODT ROUTINES
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                                                               H 12
VTST       CREATED BY  MACRO  ON 29-JUN-85 AT 03:09       PAGE 3
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE             REFERENCES

$DEVHD  = ******  GX       6-448
$FMASK  = ******  GX       6-191     6-446     6-455     6-457    *6-459
$GCLEP  = ******  GX       6-468
$GETAD  = ******  GX       6-284
$GETRC  = ******  GX       6-276
$GETRV  = ******  GX       6-269
$PARHD  = ******  GX       6-450
$PUTAD  = ******  GX       6-292
$PUTRC  = ******  GX       6-299     6-447     6-449     6-451     6-453
$RADDR  = ******  GX      *6-267    *6-274    *6-282    *6-290    *6-297    *6-447    *6-449    *6-451    *6-453
$RBLCK  = ******  GX      *6-281    *6-289
$RSIZE  = ******  GX      *6-268    *6-275    *6-283    *6-291    *6-298     6-305     6-334     6-363
$VFLAG  = ******  GX       6-186    *6-190    *6-438     6-441
$VIMI   = ******  GX       6-194
$VIMO   = ******  GX       6-458
$VTST     000036 RG       #6-182
$$$     = 001656 R        #6-184     6-184    #6-210     6-210    #6-211     6-211    #6-212     6-212    #6-213
                           6-213    #6-214     6-214    #6-215     6-215    #6-216     6-216    #6-217     6-217
                          #6-218     6-218    #6-219     6-219    #6-220     6-220    #6-221     6-221    #6-222
                           6-222    #6-223     6-223    #6-262     6-262    #6-267     6-267    #6-268     6-268
                          #6-274     6-274    #6-275     6-275    #6-281     6-281    #6-282     6-282    #6-283
                           6-283    #6-289     6-289    #6-290     6-29     #6-291     6-291    #6-297     6-297
                          #6-298     6-298    #6-307     6-307    #6-308     6-308    #6-314     6-314    #6-339
                           6-339    #6-340     6-340    #6-346     6-346    #6-362     6-362    #6-368     6-368
                          #6-382     6-382    #6-389     6-389    #6-392     6-392    #6-396     6-396    #6-399
                           6-399    #6-404     6-404    #6-405     6-405    #6-409     6-409    #6-414     6-414
                          #6-417     6-417    #6-421     6-421    #6-424     6-424    #6-429     6-429    #6-430
                           6-430    #6-434     6-434
$$$ARG  = 000003          #5-59      5-59     #5-59      5-59     #5-59      5-59     #5-59      5-59      5-59
$$$GLB  = ******           5-59
$$$LEN  = 000011          #6-210     6-210    #6-211     6-211    #6-212     6-212    #6-213     6-213    #6-214
                           6-214    #6-215     6-215    #6-216     6-216    #6-217     6-217    #6-218     6-218
                          #6-219     6-219    #6-220     6-220    #6-221     6-221    #6-222     6-222    #6-223
                           6-223    #6-262     6-262    #6-307     6-307    #6-308     6-308    #6-314     6-314
                          #6-339     6-339    #6-340     6-340    #6-346     6-346    #6-362     6-362    #6-368
                           6-368    #6-382     6-382    #6-392     6-392    #6-396     6-396    #6-399     6-399
                          #6-409     6-409    #6-417     6-417    #6-421     6-421    #6-424     6-424    #6-434
                           6-434
.CRAVL  = ******  GX       6-453
.DEVHD  = ******  GX       6-449
.FMASK  = ******  GX       6-447
.PARHD  = ******  GX       6-451
```

```
                                                               I 12
VTST       CREATED BY  MACRO  ON 29-JUN-85 AT 03:09       PAGE 4
```

```
VUMR       CREATED BY  MACRO  ON 29-JUN-85 AT 03:09     PAGE 2

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME        REFERENCES

CALL               6-104      6-105      6-108      6-137      9-238      9-241      9-243      9-261      10-292     10-318
CRASH              9-270
GETAD             #5-58
GETRV             #5-58       10-292     10-318
PUTAD             #5-58
PUTRC             #5-58       6-137      9-261
RAND              #6-137      6-137     #9-261      9-261     #10-292     10-292    #10-318     10-318
RETURN             6-141      9-265      10-320
SCBDF$            #5-58       5-63
SWTCHI            #5-58       6-139      9-263      10-291     10-294
SWTCHO            #5-58       6-135      6-140      9-251      9-264
```

```
     92                                         .SBTTL  LOCAL DATA
     93
     94                                 ;***
     95                                 ; LOCAL DATA
     96                                 ;***
     97
     98 000000                                  .PSECT  DATA,D
     99
    100
    101                                         .IF DF  R$$MPL                                          ;[MP01]
    102                                 ICBI    == 16                   ; I-SPACE ICB SIZE              ;[MP01]
    103                                 CETNAM: .ASCIZ  /CETAB/         ; CETAB FILE NAME               ;[MP01]
    104
    105                                 CTBLK:  .WORD   0,0             ; CONTROLLER SYMBOL NAME        ;[MP01]
    106                                 CTBSYM: .RAD50  "$CTXXX"                                        ;[MP01]
    107                                 CTVAL:  .BLKW   1                                               ;[MP01]
    108                                 STBEXT: .ASCIZ  /STB/           ; FILE EXTENSION NAME           ;[MP01]
    109                                         .ENDC                                                   ;[MP01]
    110
    111 000000                          $PNAME::.BLKW   2               ; PARTITION NAME                ;[MP01]
    112
    113                                 ;
    114                                 ; KRB BUFFER FOR RSX11M+ SYATEMS
    115                                 ;
    116
    117                                         .IF DF  R$$MPL                                          ;[MP01]
    118                                 KRBMP:  .BLKB   70              ; KRB BLOCK                     ;[MP01]
    119                                 CPU:    .BLKW   1               ; CPU NUMBER                    ;[MP01]
    120                                 ICBVL:  .BLKW   1               ; SAVED ICB ADDRESS             ;[MP01]
    121                                 NXICB:  .BLKW   1               ; NEXT ICB                      ;[MP01]
    122                                         .ENDC
    123
    124                                 ;
    125                                 ; READY LINES ON THIS CONTROLLER
    126                                 ;
    127 000004                          UNITS:  .BLKW   1
    128
    129                                 ;
    130                                 ; UNLOAD BLOCK BIAS AND ADDRESS
    131                                 ;
    132 000006                          UBIAS:  .BLKW   1               ; START OF BLOCK ADDRESS UNMAPPED
    133 000010                          UBA:    .BLKW   1
    134
    135                                 ;
    136                                 ; COPY OF UNLOAD BLOCK
    137                                 ;
    138 000012                          BLOCK:  .BLKW   32.
    139
    140                                 ;
    141                                 ; POINTER INTO UNLOAD BLOCK COPY
    142                                 ;
    143 000112                          UPTR:   .BLKW   1
    144
    145                                 ;
    146                                 ; BIAS OF BLOCK TO DEALLOCATE (IN 'MICUNL')
    147                                 ;
    148 000114                          KMCB:   .BLKW   1
```

```
                                                   .SBTTL  $CLIN - CLEAR LIN
                                           ;+
                                           ; $CLIN - CLEAR LINE
                                           ;
                                           ; THIS ROUTINE IS CALLED TO DE-ALLOCATE THE DLC AND DDM LINE TABLES AND TO
                                           ; REMOVE THE ASSOCIATED PROCESSES FROM CORE IF THEY HAVE NOT BEEN BROUGHT IN
                                           ; VIA "SET PROCESS" AND IF THIS IS THE LAST SLT REFERENCE TO THE PROCESS.
                                           ; IF THE LAST LINE ON A KMC HAS BEEN CLEARED, THAT KMC WILL BE REMOVED
                                           ; FROM THE "LOADED KMC'S" LIST.
                                           ;
                                           ; INPUTS:
                                           ;       $SLTA=SLT ADDRESS
                                           ;
                                           ; OUTPUTS:
                                           ;       LF.RDY!LF.UNL CLEARED FROM L.FLG
                                           ;       L.DDS AND L.DLS ZEROED
                                           ;
                                           ;-
 001610                                    $CLIN::
 001610     016705  000000G                        MOV     $SLTA,R5         ; GET SLT ADDRESS

 001614     042765  000000C 000000G        XLIN:   BIC     #LF.RDY!LF.ACT!LF.ENA,L.FLG(R5) ; MARK LINE NOT READY AND INAC
 001622                                            CALL    SCAN             ; SCAN SLT FOR OTHER ACTIVITY
 001626     116504  000000G                        MOVB    L.DDM(R5),R4     ; GET DDM PDV INDEX
 001632                                            CALL    GETPDV           ; GET PDV ADDRESS
 001636     032764  000000G 000000G                BIT     #ZF.PSE,Z.FLG(R4) ; IS THIS DLM?
 001644     001402                                 BEQ     15$              ; BR IF NO
 001646                                            CALLR   20$              ; JUMP TO 20$
 001652                                    15$:

                                                   .IF DF  R$$MPL                                              ;[MP01]

                                                   SAVRG   <R2,R4,R5>       ; SAVE REGS
                                                   MOV     $NAME,CTBSYM+2   ; FORM SYMBOL NAME TO LOOK FOR "$CTXXX"
                                                   MOV     #$DFUIC,R0       ; GET THE COMM EXEC UIC
                                                   MOV     #CETNAM,R1       ; FORM ASCIZ FILE SPEC "[XX,5X]CETAB.STB"
                                                   MOV     #STBEXT,R2       ;        ...
                                                   CALL    $AZFS            ;        ...
                                                   MOV     #CTBLK,R0        ; GET SYMBOL BLOCK TO LOOK FOR
                                                   CALL    $SYMBL           ; LOOK FOR IT

                                                   SWTCHI                   ; SET DEFAULT BUFFERS
                                                   SWTCHO                   ;      ...
                                                   CLR     R1               ; INIT REG
                                                   BISB    L.CTL(R5),R1     ; COPY CONTROLLER NUMBER
                                                   ASL     R1               ; CONVERT TO WORD OFFSET
                                                   ADD     CTVAL,R1         ; CALCULATE FIRST CTB ADDRESS
                                                   ADD     #L.KRB,R1        ; POINT AT THIS LINES KRB ADDRESS OF THE CTB
                                                   ADD     #$CETAB,R1       ; CALCULATE ADDRESS IN VNP TEMPLATE
                                                   CLR     (R1)             ; INIT KRB ADDRESS
                                                   RESRG   <R5,R4,R2>       ; RESTORE REGS

                                                   .ENDC                    ;                                   [MP01]

 001652     005702                                 TST     R2               ; ARE ANY OTHER MUX LINES STILL READY?
 001654     001066                                 BNE     10$              ; IF NE, YES
 001656                                            CALL    CSRUNL           ; REMOVE CSR ADDRESS FROM LIST
```

```
                                                   .SBTTL  ZDDS - CLEAR DDM LINE TABLE ADDRESS
                                         ;
                                         ; RE-SCAN SLT'S AND CLEAR DDM LINE TABLE ADDRESS WORD(S)
                                         ;

 003354  016702  000000G                 ZDDS:    MOV     $SLTMAB,R2          ; GET SLT MAPPING TABLE ADDRESS
 003360  016701  000000G                          MOV     $SLTNM,R1           ; GET NUMBER OF SLT'S
 003364  012200                          10$:     MOV     (R2)+,R0            ; GET NEXT SLT
 003366  126065  000000G 000000G                  CMPB    L.DDM(R0),L.DDM(R5) ; SAME DEVICE AS OURS?
 003374  001016                                   BNE     20$                 ; IF NE, NO
 003376  126065  000000G 000000G                  CMPB    L.CTL(R0),L.CTL(R5) ; SAME CONTROLLER?
 003404  001012                                   BNE     20$                 ; IF NE, NO
 003406  005060  000000G                          CLR     L.DDS(R0)           ; ZERO DDM LINE TABLE ADDRESS
 003412  005060  000000G                          CLR     L.KRBA(R0)          ; ZERO KRB ADDRESS
 003416  126060  000000G 000000G                  CMPB    L.DDM(R0),L.DLC(R0) ; COMBINATION DDM/DLC?
 003424  001002                                   BNE     20$                 ; IF NE, NO
 003426  005060  000000G                          CLR     L.DLS(R0)           ; ZERO DLC LINE TABLE ADDRESS
 003432  005301                          20$:     DEC     R1
 003434  001353                                   BNE     10$                 ; SCAN ENTIRE SLT
 003436                                           RETURN

```

```
                                                                   I  1
VRQB      CREATED BY  MACRO  ON 30-JUL-85 AT 10:55       PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE           REFERENCES

AB      = ******  GX       19-626     19-627     34-1259    34-1260
CCEX      002536 R         6-126     #26-947
CEXLD     000176 R         6-117      6-121      6-127      6-130      6-135      6-138      6-140      6-142      6-146
                           6-160     #12-353
CHKLF     001264 R         6-123     #20-667
CHKNOD    000234 R         6-136     #14-393
CHKPP     000552 R         6-119     #16-501
CLIN      002472 R         6-132     #25-920
CLQREM  = ******  GX       35-1312
CLRPP     001532 R         6-115      6-124      6-144     #21-727
CM$SET  = ******  GX       14-406
CPRO      001672 R         6-129     #24-795
DEGNTI    003640 R       #35-1302
DEV7      003424 R         8-224     #33-1225
DSIZE   = ******  GX       26-969
DV.MNT  = 100000           24-873
D.DSP     000012          *24-882
D.PCB     000034           24-858     24-864    *24-883     24-885
D.UCB     000002           24-866
D.UCBL    000010           24-870     24-878
D.UNIT    000006           24-867     24-880
ERRBUF    000400 R        #8-202      30-1139
ERRMSG    000370 R        #8-200      30-1153
FE.CEX  = ******  GX       11-319     11-337     12-354     26-948
FL.CEX  = ******  GX       11-318
FL.DDM  = ******  GX       15-474
FL.DLC    ******  GX       15-474     24-889
FL.ECH    ******  GX       7-192      9-280
FL.FDX  = ******  GX       7-190
FL.HDX  = ******  GX       7-191
FL.KMX  = ******  GX       19-608
FL.LLC  = ******  GX       15-474     16-510     17-536     17-543     24-807
FL.LMC  = ******  GX       19-600
FL.MUX  = ******  GX       19-604     29-1107
FL.TOP  = ******  GX       7-193      9-280
FL.WRN  = ******  GX       9-280
FMAX    = 000037          #6-110      9-273
FMT7      002352 R         8-209      8-210      8-211      8-212      8-213      8-214      8-215      8-216      8-217
                           8-218      8-219      8-220      8-221      8-222      8-224      8-226      8-227      8-228
                           8-229      8-230      8-231      8-232      8-233      8-234      8-235      8-236      8-237
                          #8-242
FMT7A     002353 R         8-223     #8-243
FMT7B     002366 R         8-225     #8-244
FM.7    = 000000          #8-209     #8-210     #8-211     #8-212     #8-213     #8-214     #8-215     #8-216     #8-217
                          #8-218     #8-219     #8-220     #8-221     #8-222     #8-224     #8-226     #8-227     #8-228
                          #8-229     #8-230     #8-231     #8-232     #8-233     #8-234     #8-235     #8-236     #8-237
FM.7A   = 000000          #8-223
FM.7B   = 000000          #8-225
FTAB      000000 R        #6-78       6-110      9-277
F.NRBD  = ******  GX      *9-288     *24-884
F.RSIZ  = ******  GX      *24-870    *24-885    *26-969    *26-996    *35-1303   *35-1313
F.URBD  = ******  GX      *9-287     *24-869
```

```
                                                                   J  1
VRQB      CREATED BY  MACRO  ON 30-JUL-85 AT 10:55       PAGE 2
```

```
    111
    112 000134                      90$:    SWTCHO
    113 000142                              SWTCHI
    114 000150                              RETURN
```

```
                                      .SBTTL CLEAR COMMAND

                                      ;         Keyword         Next       Action        Mask (BIS)      Flag to
                                      ;         Type, or        State      Routine       Value           Mask
                                      ;         Substate

                                ;
                                ; CLEAR COMMAND
                                ;
        000000                        STATE$   CLE,&P$CLE
        000000                         TRAN$   !IDALI,          CLAL,         ,          LX.ALI,         RQBST
        000000                         TRAN$   !IDALAL,         CLALAL,       ,          LX.ALI,         RQBST
        000000                         TRAN$   !IDKNAL,         CLAL,         ,          LX.ALI,         RQBST

        000000                         TRAN$   !IDCIR,          CLCI,         ,          LX.CIR,         RQBST
        000000                         TRAN$   !IDKNCI,         CLCI,         ,          LX.CIR,         RQBST
        000000                         TRAN$   !IDLIN,          CLLI,      $XMISS,       LX.LIN,         RQBST
        000000                         TRAN$   !IDKNLI,         CLLI,         ,          LX.LIN,         RQBST

        000000                         TRAN$   !IDLOG,          CLLO,         ,          LX.LOG,         RQBST
        000000                         TRAN$   !IDKNLO,         CLLO,         ,          LX.LOG,         RQBST
        000000                         TRAN$   !IDNOD,          CLNO,         ,          LX.NOD,         RQBST
        000000                         TRAN$   !IDOBJ,          CLOB,         ,          LX.OBJ,         RQBST
        000000                         TRAN$   !IDKNOB,         CLOB,         ,          LX.OBJ,         RQBST
        000000                         TRAN$   !IDPRO,          CLPR,         ,          LX.PRO,         RQBST
        000000                         TRAN$   !IDSYS,          CLSY,         ,          LX.CEX,         RQBST
                                ;
                                ; CLEAR ALIAS
                                ;
                                ;     ALL ALIASES
                                ;
        000000                        STATE$   CLALAL
        000000                         TRAN$   &K$ALL,          DONE
        000000                         TRAN$   $LAMDA,          DONE
                                ;
                                ;     ALIAS, KNOWN ALIASES
                                ;
        000000                        STATE$   CLAL
        000000                         TRAN$   !OPSCO,          CLAL,         ,          OP$SCO,         $OPTON
        000000                         TRAN$   &K$ALL,          CLAL
        000000                         TRAN$   $LAMDA,          DONE
                                ;
                                ; CLEAR CIRCUIT
                                ;

        000000                        STATE$   CLCI
        000000                         TRAN$     &K$ALL,        DONE
        000000                         TRAN$     $LAMDA,        DONE

                                ;
                                ; CLEAR LINE
                                ;

        000000                        STATE$   CLLI
        000000                         TRAN$     &K$ALL,        DONE
        000000                         TRAN$     $LAMDA,        DONE

```

```
    948 000000                                      STATE$  OBJID
    949 000000                                       TRAN$  $DNUMB,         $EXIT,      $OTYP
    950                                  ;
    951                                  ; PROCESS
    952                                  ;
    953 000000                                      STATE$  PROID
    954 000000                                       TRAN$  $RAD50,         $EXIT,      $PNAM
```

```
                                        ;
                                        ; FINISH OFF THE STATE TABLES
                                        ;
       000000                                STATE$
```

```
VSTB       CREATED BY  MACRO  ON 29-JUN-85 AT 03:03     PAGE 4
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE              REFERENCES

OP$NLI  = 000002           #5-62
OP$NOD  = 000020           #5-62
OP$OUT  = 002000           #5-62
OP$OWN  = 000020           #5-62
OP$PAR  = 000010           #5-62
OP$PRI  = 000010           #5-62
OP$RET  = 010000           #5-62
OP$RPA  = 000010           #5-62
OP$SCO  = 000001           #5-62
OP$SEG  = 000400           #5-62
OP$SER  = 000400           #5-62
OP$SIN  = 000010           #5-62
OP$STA  = 000004           #5-62
OP$TPA  = 000020           #5-62
OP$TRI  = 000040           #5-62
OP$UCS  = 000020           #5-62
OP$USE  = 000002           #5-62
OP$VEC  = 000040           #5-62
OP$VER  = 000040           #5-62
OP$XAC  = 000001           #5-62
OP$XPR  = 000002           #5-62
OP$X5S  = 000004           #5-62
OP$X9S  = 000010           #5-62
PARTAB    000000 RG        #6-108
PEMSTA    000040 R         #19-1640
P$ACT   = ******  G          5-71
P$AID   = ******  G          5-71
P$CLCI  = ******  G          5-71
P$CLE   = ******  G          5-71
P$CLLI  = ******  G          5-71
P$CLLO  = ******  G          5-71
P$CLNO  = ******  G          5-72
P$CLOB  = ******  G          5-72
P$COP   = ******  G          5-72
P$COS   = ******  G          5-72
P$CRID  = ******  G          5-72
P$CSR   = ******  G          5-72
P$DEA   = ******  G          5-73
P$DES   = ******  G          5-73
P$DST   = ******  G          5-73
P$DTE   = ******  G          5-73
P$DUP   = ******  G          5-73
P$EVE   = ******  G          5-73
P$GRP   = ******  G          5-73
P$HOS   = ******  G          5-74
P$HTM   = ******  G          5-74
P$LGID  = ******  G          5-74
P$LIST  = ******  G          5-74
P$LNID  = ******  G          5-74
P$LOC   = ******  G          5-74
P$LOST  = ******  G          5-75
P$LTM   = ******  G          5-75
```

```
VSTB       CREATED BY  MACRO  ON 29-JUN-85 AT 03:03     PAGE 5
```

```
                  000001                LN.OPE==1        ; SUBSTATE = OPEN, UNSPECIFIED
                  000002                LN.REF==2        ; SUBSTATE = OPEN, REFLECTING
                  000003                LN.LOO==3        ; SUBSTATE = OPEN, LOOPING
                  000004                LN.LOA==4        ; SUBSTATE = OPEN, LOADING
                  000005                LN.DUM==5        ; SUBSTATE = OPEN, DUMPING
                  000006                LN.TRI==6        ; SUBSTATE = OPEN, TRIGGERING
                  000000                        .=0
                                        ;
                                        ;       MULTIPOINT FLAG TABLE OFFSETS AND FLAG BYTE DEFINITIONS
                                        ;
          000000                        S.FLG:: .BLKB    1      ; FLAGS BYTE
          000001                        S.COST::         .BLKB    1       ; TRIBUTARY COST FOR ROUTING TASK
          000002                        S.NMST::         .BLKB    1       ; NETWORK MANAGEMENT TRIBUTARY STATE/SUBSTATE
          000003                        S.OWNR::         .BLKB    1       ; TRIBUTARY OWNER PROCESS INDEX
          000004                        S.LEN::                 ; LENGTH OF TRIBUTARY EXTENSION
                                        ;
                                        ;       FLAG BYTE DEFINITIONS
                                        ;
                  000200                SF.ACT==000200   ; TRIBUTARY IS ACTIVE
                  000100                SF.ENA==000100   ; TRIBUTARY IS MARKED FOR ENABLE AT INITIALIZATION TIME
                  000040                SF.UNL==000040   ; TRIBUTARY IS MARKED FOR UNLOAD
                  000040                SF.MFL==000040   ; TRIBUTARY IS MARKED FOR LOAD (DATA LINK MAPPING ONLY)
                  000020                SF.PAC==000020   ; TRIBUTARY WAS PREVIOUSLY ACTIVE
                                                                ; (BEFORE RECONFIGURATION)
                  000010                SF.REA==000010   ; TRIBUTARY IS REASSIGNABLE
                  000004                SF.LPB==000004   ; TRIBUTARY IS IN LOOPBACK
                  000002                SF.SVC==000002   ; DLM CIRCUIT IS AN SVC
                  000001                SF.SER==000001   ; CIRCUIT HAS SERVICE DISABLED

     66
     67                                 ;***
     68                                 ; LOCAL SYMBOLS
     69                                 ;***
     70                                 ;
     71                                 ; TEMPLATE PROCESSING FUNCTION CODES
     72                                 ;
     73                                 ; WARNING - DON'T CHANGE THESE CODES WITHOUT CHANGING TMPEX2 AND TMPEX4.
     74                                 ;
     75 000004                                  .ASECT
     76                   000001                .=1                     ; 0=END OF FILE
     77 000001                          .ADDRB::.BLKB    1
     78 000002                          .ADDRW::.BLKB    1
     79 000003                          .BLKB:: .BLKB    1
     80 000004                          .BLKW:: .BLKB    1
     81 000005                          .BYTE:: .BLKB    1
     82 000006                          .CORE:: .BLKB    1
     83 000007                          .CSR::  .BLKB    1
     84 000010                          .CTIM:: .BLKB    1
     85 000011                          .DVCHA::.BLKB    1
     86 000012                          .EOF::  .BLKB    1
     87 000013                          .FILE:: .BLKB    1
     88 000014                          .INT::  .BLKB    1
     89 000015                          .LIBR:: .BLKB    1
     90 000016                          .LINKS::.BLKB    1
     91 000017                          .LFILE::.BLKB    1
     92 000020                          .LSTHD::.BLKB    1
     93 000021                          .LTAB:: .BLKB    1
```

```
                        .SBTTL  MACRO DEFINITIONS
                ;**
                ; LIBRARY MACROS
                ;**
                        .MCALL  EMSG$,EMSG$R,NTLER$,PUTAD,SWTCHO,PUTRC
                        .MCALL  ERMSG$,ERBLK$,ERRPT$
                        .ENABL  LC

                        .ENABL  LC
```

```
    527                                                 .SBTTL  $VFDDM - VERIFY DDM
    528                                         ;+
    529                                         ; $VFDDM - VERIFY DDM ACTION ROUTINE
    530                                         ;
    531                                         ; INPUTS:
    532                                         ;       NONE
    533                                         ;
    534                                         ; OUTPUTS:
    535                                         ;       ERROR UNLESS DDM BEING LOADED
    536                                         ;-
    537 001344                                  $VFDDM::
    538 001344  032767  000000G 000000G                 BIT     #FL.DDM,$FLAGS  ; IS THIS A DDM?
    539 001352  001003                                  BNE     10$             ; IF NE, YES
    540 001354                                          EMSG$   21              ; DDM OPERATOR IN DLC/LLC TEMPLATE
    541 001362                                  10$:    RETURN
```

```
     54                                             .SBTTL  MACRO DEFINTIONS
     55                                     ;***
     56                                     ; LIBRARY MACROS
     57                                     ;***
     58                                             .MCALL  EMSG$R,NTLER$,NKRDF$,EMSG$,GETRV,SAVRG,RESRG
     59
     60                                             .IF NDF R$$MPL                                          ;[MP01]
     61 000000                                      NKRDF$                          ; DEFINE NETWORK KRB OFFSETS
     62                                             .ENDC                                                   ;[MP01]
     63
     56                                         .MCALL  GETRV,GETRC,GETAD,PUTAD,PUTRC,SWTCHO,SVTK$S
     57                                         .MCALL  QIOW$,DIR$,GCL$
     58
     59 000000                          MSG:    QIOW$   IO.WLB,2,2,,,,<0,0,40>
     60         000014'                         MADDR = MSG+Q.IOPL
     61         000016'                         MLGTH = MSG+Q.IOPL+2
     62         000020'                         MCTL = MSG+Q.IOPL+4
     63                                 ;
     64                                 ; CONSTANTS
     65                                 ;
     66         000040                          LFCR = 40                       ; LINE FEED/CARRIAGE RETURN
     67         000044                          LF = 44                         ; LINE FEED
     68         000060                          DS = 60                         ; DOUBLE SPACE
     69         000053                          CR = 53                         ; CARRIAGE RETURN
     70         000000                          NO = 0                          ; NO CARRIAGE CONTROL
     71         015010                          OB = 15010                      ; OCTAL BINARY
     72         015012                          DB = 15012                      ; DECIMAL BINARY
     73         031010                          OW = 31010                      ; OCTAL WORD
     74         031012                          DW = 31012                      ; DECIMAL WORD
     75
     76                                         .MACRO  TYPES   TEXT,CTL
     77                                         .NCHR   $$$LEN,<TEXT>
     78                                         .PSECT  $$$VDT,D
     79                                         $$$=.
     80                                         .ASCII  %TEXT%
     81                                         .PSECT
     82                                         MOV     #$$$LEN,MLGTH
     83                                         MOV     #$$$,MADDR
     84                                         .IF NB  CTL
     85                                         MOV     #CTL,MCTL
     86                                         .IFF
     87                                         MOV     #LFCR,MCTL
     88                                         .ENDC
     89                                         DIR$    #MSG
     90                                         .ENDM   TYPES
     91
     92                                         .MACRO TYPEB ADDR,CTL,LEN
     93                                         MOV     ADDR,MADDR
     94                                         MOV     LEN,MLGTH
     95                                         .IF NB CTL
     96                                         MOV     #CTL,MCTL
     97                                         .IFF
     98                                         MOV     #LFCR,MCTL
     99                                         .ENDC
    100                                         DIR$    #MSG
    101                                         .ENDM
    102
    103                                         .MACRO  TYPEN NUM,CTL,CON
    104                                         .IF NB CON
    105                                                 .IF DIF <NUM>,<R1>
    106                                                 .IF IDN <CON>,<OB>
    107                                                 CLR     R1
    108                                                 BISB    NUM,R1                  ; GET NUMBER TO CONVERT
    109                                                 .IFF
    110                                                 .IF IDN <NUM>,<DB>
    111                                                 CLR     R1
    112                                                 BISB    NUM,R1                  ; GET NUMBER TO CONVERT
```

```
                                                                                  I 12
VTST        CREATED BY  MACRO  ON 29-JUN-85 AT 03:09      PAGE 4
MACRO CROSS REFERENCE                                     CREF    04.00

MACRO NAME      REFERENCES

CALL              6-184     6-188     6-194     6-203     6-267     6-267     6-268     6-268     6-269     6-274
                  6-274     6-275     6-275     6-276     6-281     6-281     6-282     6-282     6-283     6-283
                  6-284     6-289     6-289     6-290     6-290     6-291     6-291     6-292     6-297     6-297
                  6-298     6-298     6-299     6-306     6-313     6-338     6-345     6-367     6-370     6-372
                  6-377     6-389     6-389     6-390     6-397     6-398     6-404     6-404     6-405     6-405
                  6-407     6-414     6-414     6-415     6-422     6-423     6-429     6-429     6-430     6-430
                  6-432     6-447     6-449     6-451     6-453     6-458     6-468     6-483
DIR$             #5-57      6-210     6-211     6-212     6-213     6-214     6-215     6-216     6-217     6-218
                  6-219     6-220     6-221     6-222     6-223    #6-257     6-257    #6-259     6-259     6-262
                  6-306     6-307     6-308     6-313     6-314     6-319     6-321     6-338     6-339     6-340
                  6-345     6-346     6-351     6-353     6-362     6-367     6-368     6-370     6-382     6-392
                  6-396     6-397     6-398     6-399     6-409     6-417     6-421     6-422     6-423     6-424
                  6-434
ENTEXT           #5-157     6-488
GCL$             #5-57      6-468
GETAD            #5-56      6-284
GETNUM           #5-141     6-267     6-268     6-274     6-275     6-281     6-282     6-283     6-289     6-290
                  6-291     6-297     6-298     6-389     6-404     6-405     6-414     6-429     6-430
GETRC            #5-56      6-276
GETRV            #5-56      6-269
MOV$             #6-257     6-257     6-257    #6-259     6-259     6-259
OFF$             #5-59
PUTAD            #5-56      6-292
PUTRC            #5-56      6-299     6-447     6-449     6-451     6-453
QDPB$            #5-59      5-59
QIOW$            #5-57      5-59
QREAD            #5-132     6-184     6-267     6-268     6-274     6-275     6-281     6-282     6-283     6-289
                  6-290     6-291     6-297     6-298     6-389     6-404     6-405     6-414     6-429     6-430
RAND             #6-269     6-269    #6-276     6-276    #6-284     6-284    #6-292     6-292    #6-299     6-299
                 #6-447     6-447    #6-449     6-449    #6-451     6-451    #6-453     6-453
RETURN            6-195     6-260     6-270     6-277     6-285     6-293     6-300     6-322     6-354     6-385
                  6-400     6-410     6-425     6-435     6-460     6-477     6-486
SVTK$S           #5-56      6-257     6-259
SWTCHO           #5-56      6-446     6-448     6-450     6-452     6-454
TYPEB            #5-92      6-319     6-321     6-351     6-353
TYPEN            #5-103     6-306     6-313     6-338     6-345     6-367     6-370     6-397     6-398     6-422
                  6-423
TYPES            #5-76      6-210     6-211     6-212     6-213     6-214     6-215     6-216     6-217     6-218
                  6-219     6-220     6-221     6-222     6-223     6-262     6-307     6-308     6-314     6-339
                  6-340     6-346     6-362     6-368     6-382     6-392     6-396     6-399     6-409     6-417
                  6-421     6-424     6-434
```

```
                                                                                  J 12
**FILE**ID**VUMP
```

```
**FILE**ID**VUNB
```

```
VV      VV   UU      UU   NN      NN   BBBBBBBB
VV      VV   UU      UU   NN      NN   BBBBBBBB
VV      VV   UU      UU   NN      NN   BB      BB
VV      VV   UU      UU   NN      NN   BB      BB
VV      VV   UU      UU   NNNN    NN   BB      BB
VV      VV   UU      UU   NNNN    NN   BB      BB
VV      VV   UU      UU   NN  NN  NN   BBBBBBBB
VV      VV   UU      UU   NN  NN  NN   BBBBBBBB
VV      VV   UU      UU   NN    NNNN   BB      BB
VV      VV   UU      UU   NN    NNNN   BB      BB
  VV  VV     UU      UU   NN      NN   BB      BB       ....
  VV  VV     UU      UU   NN      NN   BB      BB       ....
    VV       UUUUUUUUUU   NN      NN   BBBBBBBB         ....
    VV       UUUUUUUUUU   NN      NN   BBBBBBBB         ....


LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```

                                      ;
                                      ; LOOP COUNTER
                                      ;
000116                                COUNT:  .BLKW   1

                                      ;
                                      ; "CLEAR LINE" COUNTERS
                                      ;
000120                                DLCAC:  .BLKW   1               ; DLC ACTIVITY COUNT
000122                                DDMAC:  .BLKW   1               ; DDM ACTIVITY COUNT

000124  055251  054374                NTIR:   .RAD50  /NTINIT/        ; NTINIT RAD50 VALUE

                                      ;
                                      ; NONSENSE INTERRUPT TABLE (RSX11M ONLY)
                                      ;
000130  000000G                       NSITAB: .WORD   .NS0
000132  000000G                               .WORD   .NS1
000134  000000G                               .WORD   .NS2
000136  000000G                               .WORD   .NS3
000140  000000G                               .WORD   .NS4
000142  000000G                               .WORD   .NS5
000144  000000G                               .WORD   .NS6
000146  000000G                               .WORD   .NS7
                                      ;
                                      ; PARTITION LINE TABLE OFFSETS
                                      ;
000150                                ICBLK:  .BLKW   10.             ; SPACE FOR ICB
000174                                ICBSAV: .BLKW   1               ; START ADDRESS OF ICB
000176                                ICBSZ:  .BLKW   1               ; SIZE OF AN ICB
000200                                PRLTA:  .BLKW   1               ; FLAG WORD

        000001                        PR.LTA=1                        ; SET IF PARTITION LINE TABLE USED
        000002                        PR.SEC=2                        ; SET IF NOT FIRST TIME THRU VECSC

                                      ;
                                      ; PDV ADDRESS FOR PARTITION LINE TABLE DE-ALLOCATOR
                                      ;
000202                                ..PDV:  .BLKW   1               ; IF ZERO, DISCARD INSTEAD OF DE-ALLOCATE

000000                                        .PSECT
```

```
                                ;
                                ; FIND LINE TABLE
                                ;
001662  016503  000000G                 MOV     L.KRBA(R5),R3   ; GET KRB ADDRESS
001666  001001                          BNE     3$              ; IF OK - BRANCH                        ;[MP01]
001670                                  CRASH                   ; IF ZERO, NO KRB ?
001672                          3$:
                                        .IF DF  R$$MPL                                                  ;[MP01]
                                        SWTCHI  #KRBMP          ; M+ KRB BUFFER                         ;[MP01]
                                        ADD     #K.PRM,R3       ; BACKUP POINTER TO ACTUAL KRB START    ;[MP01]

                                        .ENDC                                                           ;[MP01]

001672                                  GETRV   R3,#$KRBLN      ; READ IN THE KRB

                                        .IF DF  R$$MPL                                                  ;[MP01]
                                        MOV     #KRBMP,R3       ; POINT TO M+ KRB OFFSET ZERO           ;[MP01]
                                        SUB     #K.PRM,R3       ; ...                                   ;[MP01]
                                        CLR     -(SP)           ; CLEAR A WORD                          ;[MP01]
                                        BISB    K.VCT(R3),(SP)  ; GET VECTOR ADDRESS/4                  ;[MP01]
                                        TST     .MPRO           ; MULTIPROCESSOR ?                      ;[MP01]
                                        BLT     2$              ; IF NO - BRANCH                        ;[MP01]
                                        MOV     K.URM(R3),R3    ; GET UNIBUS RUN MASK                   ;[MP01]
                                        MOV     #-1,-(SP)       ; INIT CPU NUMBER                       ;[MP01]
                                1$:     INC     (SP)            ; COUNT ONE PROCESSOR                   ;[MP01]
                                        ASR     R3              ; CORRECT CPU ?                         ;[MP01]
                                        BCC     1$              ; IF NO - BRANCH                        ;[MP01]
                                        CLC                     ; CLEAR C-BIT                           ;[MP01]
                                        MOV     (SP)+,$BFR      ; COPY CPU NUMBER                       ;[MP01]
                                        ASL     $BFR            ; CONVERT TO WORD INDEX                 ;[MP01]
                                        ADD     .K6TAB,$BFR     ; GET ADDRESS OF CPU PARTITION          ;[MP01]
                                        SWTCHI  #$CPBIA         ; SET INPUT BUFFER                      ;[MP01]
                                        GETRV   $BFR,#2         ; READ BIAS                             ;[MP01]
                                        MOV     $CPBIA,$RBLCK   ; SET UP CPU BIAS                       ;[MP01]
                                        BR      4$              ; CONTINUE                              ;[MP01]
                                2$:     CLR     $RBLCK          ; VECTOR BIAS = 0 IF NON-MULTIPRO       ;[MP01]
                                4$:     MOV     (SP)+,R3        ; POP VECTOR ADDRESS                    ;[MP01]

                                        .IFF                                                            ;[MP01]
001710  116703  000001G                 MOVB    K.VCT+$BFR,R3   ; GET THE VECTOR ADDRESS/4

                                        .ENDC                                                           ;[MP01]

                                        .IF DF  R$$MPL                                                  ;[MP01]
                                        SWTCHI                                                          ;[MP01]

                                        .ENDC                                                           ;[MP01]

001714  006303                          ASL     R3              ; CONVERT TO REAL VECTOR ADDRESS
001716  006303                          ASL     R3              ; ...
001720                                  CALL    FNDADD          ; FIND THE INTERRUPT BLOCK IN THE LINE TABLE
001724  116504  000000G                 MOVB    L.DDM(R5),R4    ; GET THE DDM PDV INDEX
001730                                  CALL    FLT16           ; FIND LINE TABLE IN UNLOAD DATA
                                ;
                                ; FIND THE PDV BIAS
                                ;
001734                                  CALL    GETPDV          ; CONVERT TO PDV ADDRESS
```

```
                                                  .SBTTL  CSRUNL - REMOVE CSR UNLOAD ENTRY
                                        ;+
                                        ;         CSRUNL   - REMOVE ENTRY IN CSR CHECK LOAD BLOCK
                                        ;
                                        ; INPUTS:
                                        ;         $SLTA    - SYSTEM LINE TABLE ADDRESS
                                        ;
                                        ; OUTPUT:
                                        ;         ENTRY IS REMOVED FROM BLOCK, BUT IF BLOCK IS ALL ZEROS
                                        ;         BLOCK IS NOT REMOVED FROM LIST SINCE LINE WILL PROBABLY
                                        ;         BE RELOADED ANYWAY.
                                        ;-

 003440                                 CSRUNL: CALL    $SAVAL              ; SAVE ALL REGISTERS
 003444  012700  000000C                        MOV     #$NTLHB+.CXCSR,R0 ; POINT AT LISTHEAD
 003450  012701  000004                         MOV     #4,R1               ; SIZE OF ENTRIES
 003454  011002                                 MOV     (R0),R2             ; LISTHEAD VALUE
 003456  001442                                 BEQ     101$                ; IMPOSSIBLE
 003460  012702  000002G                        MOV     #$UNBLK+2,R2        ; SET STARTING ADDRESS
 003464  160102                                 SUB     R1,R2               ; ...
 003466  016705  000000G                        MOV     $SLTA,R5            ; GET THE SLT ADDRESS

 003472                                 10$:    CALL    $VUNB               ; POINT AT NEXT ENTRY
 003476  103431                                 BCS     90$                 ; MUST HAVE BEEN DUPLICATE
 003500  011203                                 MOV     (R2),R3             ; GET SLT ADDRESS FROM THE CSR LIST
 003502  001773                                 BEQ     10$                 ; DON'T TRY ZERO ENTRIES
 003504  166703  000000G                        SUB     $CETDF,R3           ; REMOVE ADDRESSING OFFSET
 003510  126365  000000G 000000G                CMPB    L.DDM(R3),L.DDM(R5) ; IS IT THE SAME DEVICE ?
 003516  001365                                 BNE     10$                 ; NO ..
 003520  126365  000000G 000000G                CMPB    L.CTL(R3),L.CTL(R5) ; IS IT THE SAME CONTROLLER ?
 003526  001361                                 BNE     10$                 ; NO ..

 003530  005022                                 CLR     (R2)+               ; ZERO OUT ENTRY
 003532  005022                                 CLR     (R2)+               ; ...
 003534                                         SWTCHO  #$UNBLK             ; POINT AT UNLOAD BLOCK
 003542                                         PUTAD   ,#64.               ; REWRITE MODIFIED BLOCK
 003554                                         SWTCHO                      ; RESET DEFAULT BUFFER
 003562                                 90$:    RETURN

                                        ;
                                        ; ERRORS
                                        ;

 003564                                 101$:   ERRPT$  ERR77,$EROUT,FATAL ; LOGIC ERROR IN CSR BLOCK
 003600                                         ERMSG$  <CSR entry missing from check list. ( Logic error. )>
 003663                                         ERBLK$  ERR77


```

```
                                                                J  1
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 2
SYMBOL  CROSS REFERENCE                                   CREF    04.00
SYMBOL  VALUE             REFERENCES

I$%AS   = ******           20-685
IBITS     000346 R        #7-190        22-742
LD.ALI  = 000030          #5-63
LD.CEX  = 000024          #5-63
LD.CIR  = 000035          #5-63
LD.LIN  = 000025          #5-63
LD.LOG  = 000031          #5-63
LD.MOD  = 000037          #5-63
LD.NOD  = 000027          #5-63
LD.OBJ  = 000032          #5-63
LD.PRO  = 000026          #5-63
LF.ACT  = ******   GX      24-848       25-926
LF.RDY  = ******   GX      22-745       23-774        24-817        24-897        25-922
LF.UNL  = ******   GX      25-928
LLIN      000746 R         6-139       #18-565
LO.ADD    000022          #5-63
LO.ADR    000052          #5-63
LO.CLS    000002          #5-63
LO.CON    000006          #5-63
LO.COP    000010          #5-63
LO.COS    000020          #5-63
L    R    000010          #5-63         5-63
LO.CTL    000032          #5-63
LO.DES    000002          #5-63         5-63
LO.DTE    000047          #5-63
LO.EVT    000004          #5-63
LO.GRO    000024          #5-63
LO.HOS    000040          #5-63
LO.HTM    000032          #5-63
LO.INA    000062          #5-63
LO.INC    000060          #5-63
LO.LDT    000046          #5-63
LO.LID    000030          #5-63         18-568
LO.LIN    000004          #5-63
LO.LTM    000034          #5-63
LO.MAC    000022          #5-63
LO.MDE    000024          #5-63
LO.NAM    000044          #5-63        *14-441
LO.NET  = 000002          #5-63
LO.NNM    000022          #5-63
LO.OFL    000012          #5-63
LO.ONM    000004          #5-63
LO.OUT    000056          #5-63
LO.OWN    000026          #5-63
LO.PAR    000010          #5-63
LO.PRI    000012          #5-63
LO.RET    000064          #5-63
LO.RPA    000030          #5-63
LO.SCR    000016          #5-63
LO.SEG    000054          #5-63
LO.SKA    000014          #5-63
LO.SKN    000014          #5-63
```

```
                                                                K  1
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 3
```

```
    116
    117
    118         000001                        .END
```

```
                        ;
                        ;  CLEAR LOGGING
                        ;
000000                          STATE$  CLLO,&P$CLLO
000000                           TRAN$   $LAMDA,        CLLOOP

000000                          STATE$  CLLOOP
000000                           TRAN$  !OPCSNM,        CLLOOP,   ,         OP$NAM,     $OPTON
000000                           TRAN$  !OPEVE,         CLLOOP,   ,         OP$EVE,     $OPTON
000000                           TRAN$  !OPSIN,         CLLOOP,   ,         OP$SIN,     $OPTON
000000                           TRAN$  !OPLIN,         CLLOOP,   ,         OP$LNE,     $OPTON
000000                           TRAN$  !OPNOD,         CLLOOP,   ,         OP$NOD,     $OPTON
000000                           TRAN$  !OPCIR,         CLLOOP,   ,         OP$CIE,     $OPTON
000000                           TRAN$  $LAMDA,         DONE,     $CKLOG
                        ;
                        ;  CLEAR NODE
                        ;
000000                          STATE$  CLNO,&P$CLNO
000000                           TRAN$   $LAMDA,        CLNOOP
                        ;
                        ;     CLEAR NODE OPTIONS
                        ;
000000                          STATE$  CLNOOP
000000                           TRAN$  &K$NAM,         CLNOOP,   ,         OP$NAM,     $OPTON
000000                           TRAN$  &K$CIR,         CLNOOP,   ,         OP$NLI,     $OPTON
000000                           TRAN$  &K$HOS,         CLNOOP,   ,         OP$HOS,     $OPTON
000000                           TRAN$  !KWRPAS,        CLNOOP,   ,         OP$RPA,     $OPTON
000000                           TRAN$  !KWTPAS,        CLNOOP,   ,         OP$TPA,     $OPTON
000000                           TRAN$  $LAMDA,         DONE
                        ;
                        ;  CLEAR OBJECT
                        ;
000000                          STATE$  CLOB
000000                           TRAN$   &K$ALL,        DONE
000000                           TRAN$   $LAMDA,        DONE
                        ;
                        ;  CLEAR PROCESS
                        ;
000000                          STATE$  CLPR
000000                           TRAN$  &K$ALL,         DONE
000000                           TRAN$  $LAMDA,         DONE
                        ;
                        ;  CLEAR SYSTEM
                        ;
000000                          STATE$  CLSY
000000                           TRAN$  &K$ALL,         DONE
000000                           TRAN$  $LAMDA,         DONE
```

```
                        .SBTTL  ENTITY OPTION SUBEXPRESSIONS

                        ;       Keyword         Next        Action      Mask (BIS)      Flag to
                        ;       Type, or        State       Routine     Value           Mask
                        ;       Substate

                ;
                ; ENTITY OPTION SUBEXPRESSIONS
                ;
                ;
                ; CONTROLLER CSR
                ;
000000                  STATE$  OPCCSR
000000                   TRAN$  &K$CON,         OPCCS1
000000                   TRAN$  $LAMDA
000000                  STATE$  OPCCS1
000000                   TRAN$  &K$CSR
000000                  STATE$   ,&P$CSR
000000                   TRAN$  $NUMBR,         $EXIT,      $CSR,       LS.OPT,         RQBST

                ;
                ; CIRCUIT
                ;
000000                  STATE$  OPCIR
000000                   TRAN$  &K$CIR
000000                  STATE$   ,&P$CRID
000000                   TRAN$  !CIRID,         $EXIT

                ;
                ; COPIES
                ;
000000                  STATE$  OPCOP
000000                   TRAN$  &K$COP
000000                  STATE$   ,&P$COP
000000                   TRAN$  &K$SNG,         $EXIT
000000                   TRAN$  $DNUMB,         $EXIT,      $COP,       OF$SMC,         RQB+LO.OFL

                ;
                ; COST
                ;
000000                  STATE$  OPCOS
000000                   TRAN$  &K$COS
000000                  STATE$   ,&P$COS
000000                   TRAN$  $DNUM8,         $EXIT,      $COST

                ;
                ; DESTINATION FOR SET ALIAS
                ;
000000                  STATE$  OPDES
000000                   TRAN$  &K$DES
000000                  STATE$  OPDESV,&P$DES
000000                   TRAN$  !DESID,         $EXIT


                ;
                ; X25-ACCESS - DESTINATION or NETWORK options                               ;[TD001]
                ;                                                                           ;[TD001]
                                                                                            ;**-2
000000                  STATE$  OPADS
000000                   TRAN$    &K$ALL,       OPDST1
000000                   TRAN$    &K$NET,       OPANT           ; Check for ... NETWORK ...     ;[TD001]
000000                   TRAN$    &K$KNO,       OPAD2           ; Check for KNOWN NETs/DESTs    ;[TD001]
```

```
   1589                                              .SBTTL INTERNAL ACTION MODULES
   1590                                      ;
   1591                                      ; INTERNAL ACTION MODULES
   1592                                      ;
   1593                                      ; BLANK PARSE MANUALLY
   1594                                      ;
   1595 000000                               BL.MAN:
   1596 000000  052767  000001  000000G              BIS     #1,..PFLAG              ;MAKE BLANKS SIGNIFICANT
   1597
   1598 000006                                       RETURN
   1599                                      ;
   1600                                      ; BLANK PARSE AUTOMATICALLY
   1601                                      ;
   1602 000010                               BL.AUT:
   1603 000010  042767  000001  000000G              BIC     #1,..PFLAG              ;MAKE BLANKS INSIGNIFICANT
   1604
   1605 000016                                       RETURN
```

```
VSTB       CREATED BY  MACRO  ON 29-JUN-85 AT 03:03     PAGE 5
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE            REFERENCES

P$MAX   = ******   G      5-75
P$MOID  = ******   G      5-75
P$MUL   = ******   G      5-75
P$MXCO  = ******   G      5-75
P$MXLI  = ******   G      5-76
P$NAD   = ******   G      5-76
P$NAM   = ******   G      5-76
P$NET   = ******   G      5-76
P$NID   = ******   G      5-76
P$NOST  = ******   G      5-76
P$OBNA  = ******   G      5-76
P$OBUS  = ******   G      5-77
P$OBVE  = ******   G      5-77
P$OID   = ******   G      5-77
P$OWN   = ******   G      5-77
P$PAR   = ******   G      5-77
P$PEN   = ******   G      5-77
P$PID   = ******   G      5-78
P$PRI   = ******   G      5-78
P$SCO   = ******   G      5-78
P$SEAL  = ******   G      5-78
P$SECI  = ******   G      5-78
P$SELI  = ******   G      5-78
P$SELO  = ******   G      5-79
P$SEMO  = ******   G      5-79
P$SENO  = ******   G      5-79
P$SEOB  = ******   G      5-79
P$SEPR  = ******   G      5-79
P$SET   = ******   G      5-79
P$SHO   = ******   G      5-79
P$SIN   = ******   G      5-80
P$SNAM  = ******   G      5-80
P$TER   = ******   G      5-80
P$TPAS  = ******   G      5-80
P$TRIB  = ******   G      5-80
P$VEC   = ******   G      5-80
P$VEST  = ******   G      5-81
P$VNP   = ******   G      5-81
P$X5ST  = ******   G      5-81
QF$ACT  = 000001         #5-62
QF$ALL  = 000002         #5-62
QF$KNO  = 000004         #5-62
QF$LOO  = 000010         #5-62
REJECT    000126 R       #21-1671
RQB     = ******   G      5-85
RQBST   = ******   G      5-85       *19-1641
STMAT1    000020 R       #18-1619
STMAT3    000030 R       #18-1623
STR.B     000134 R       #22-1678
TAB     = 000011         #5-64        22-1683
TY$CHA  = 000400         #5-62
TY$EVE  = 001000         #5-62
```

```
     94 000022                        .MPTAB::.BLKB   1
     95 000023                        .MXTAB::.BLKB   1
     96 000024                        .NODE:: .BLKB   1
     97 000025                        .PRI::  .BLKB   1
     98 000026                        .PRTOV::.BLKB   1
     99 000027                        .SCOM:: .BLKB   1
    100 000030                        .SECSR::.BLKB   1
    101 000031                        .SLNB:: .BLKB   1
    102 000032                        .SLNW:: .BLKB   1
    103 000033                        .STNB:: .BLKB   1
    104 000034                        .STNW:: .BLKB   1
    105 000035                        .UNB::  .BLKB   1
    106 000036                        .UNW::  .BLKB   1
    107 000037                        .WORD:: .BLKB   1
    108 000040                        .APR::  .BLKB   1
    109 000041                        .DPRB:: .BLKB   1
    110 000042                        .DPRW:: .BLKB   1
    111 000043                        .TIME:: .BLKB   1
    112 000044                        .MXPTB::.BLKB   1
    113 000045                        .PECHA::.BLKB   1
    114 000046                        .X3C01::.BLKB   1
    115 000047                        .X3C02::.BLKB   1
    116 000050                        .X3C03::.BLKB   1
    117 000051                        .X3C04::.BLKB   1
    118 000052                        .X3C05::.BLKB   1
    119 000053                        .X3C06::.BLKB   1
    120 000054                        .X3C07::.BLKB   1
    121 000055                        .X3C08::.BLKB   1
    122 000056                        .X3C09::.BLKB   1
    123 000057                        .X3C10::.BLKB   1
    124 000060                        .X3C11::.BLKB   1
    125 000061                        .X2CH1::.BLKB   1
    126 000062                        .X2CH2::.BLKB   1
    127 000063                        .X2CH3::.BLKB   1
    128 000064                        .X2CH4::.BLKB   1
    129 000065                        .X2CH5::.BLKB   1
    130 000000                                .PSECT
    131
    132                               ;
    133                               ; DATA BLOCK OFFSETS (POINTED TO BY $NTLPT - NTPOOL SPACE)
    134                               ;
    135                               ; WARNING - DON'T CHANGE THESE OFFSETS WITHOUT RE-BUILDING VNP AND NTINIT.
    136                               ;
    137 000066                                .ASECT
    138             000000                    .=0
    139 000000                                .BLKW   1               ; NTPOOL PCB ADDRESS (SETUP BY NTINIT)
    140 000002                                .BLKW   1               ; NTPOOL UMR BLOCK (SETUP BY NTINIT)
    141 000004                        .CXCSR::.BLKW   1               ; VNP CSR->SYSTEM LINE LIST
    142 000006                        .CXPCB::.BLKW   1               ; COMM EXEC PCB ADDRESS
    143 000010                        .CXSYM::.BLKW   3               ; PROCESS VERIFICATION SYMBOLS
    144 000016                        .CXVSM::.BLKW   3               ; VECTORED PROCESS VERIFICATION SYMBOLS
    145 000024                        .CXUNL::.BLKW   1               ; FIRST UNLOAD BLOCK ADDRESS
    146 000026                        .CXVEC::.BLKW   1               ; FIRST KMC CSR BLOCK ADDRESS
    147 000030                        .CXLB1::.BLKW   6               ; TWO LIBRARY REFERENCES
    148 000044                        .CXLB2::.BLKW   6               ;   ...
    149 000060                        .CXLBR::.BLKW   1               ; FIRST EXTRA LIBRARY BLOCK ADDRESS
    150 000062                        .CXALL::.BLKW   1               ; CETAB ALLOCATION ADDRESS
```

```
     64                                         .SBTTL  LOCAL DATA
     65                                 ;**
     66                                 ; LOCAL DATA
     67                                 ;**
     68
     69 000000                                  .PSECT  DATA,D
     70
     71                                 ;
     72                                 ; TEMPLATE FILE FLAGS
     73                                 ;
     74 000000                          $FLAG1::.BLKW   1
     75
     76         000001                  F1.BBO==1                       ; BINARY BUFFER OVERFLOWED
     77         000002                  F1.INT==2                       ; ''.INT'' OPERATOR SEEN
     78         000004                  F1.IN1==4                       ; ''.INT1'' OPERATOR SEEN
     79         000010                  F1.IN2==10                      ; ''.INT2'' OPERATOR SEEN
     80         000020                  F1.IN3==20                      ; ''.INT3'' OPERATOR SEEN
     81         000040                  F1.LFI==40                      ; ''.LFILE'' OPERATOR SEEN
     82         000100                  F1.LFL==100                     ; ''.LFLHD'' OPERATOR SEEN
     83         000200                  F1.MPL==200                     ; ''.MPLHD'' OPERATOR SEEN
     84         000400                  F1.MPT==400                     ; ''.MPTAB'' OPERATOR SEEN
     85         001000                  F1.MXT==1000                    ; ''.MXTAB'' OPERATOR SEEN
     86         002000                  F1.STE==2000                    ; SYMBOL TABLE ERROR HAS OCCURRED
     87         004000                  F1.UMR==4000                    ; ALLOCATE FROM UMR MAPPED AREA OF NTPOOL
     88         010000                  F1.VSE==10000                   ; VECTOR SETUP ERROR OCCURRED
     89         100000                  F1.ERR==100000                  ; SOME ERROR OCCURRED
     90                                                                 ; *** F1.ERR MUST BE THE SIGN BIT ***
     91
     92                                 ;
     93                                 ; TEMPLATE FILE LINE FLAGS
     94                                 ;
     95 000002                          $FLAG2::.BLKW   1
     96
     97         000001                  F2.BYT==1                       ; BYTE MODE
     98         000002                  F2.END==2                       ; ''.END'' OPERATOR SEEN
     99         000004                  F2.FIL==4                       ; VALID FILE NAME SEEN
    100         000010                  F2.LBR==10                      ; LIBRARY REFERENCE ALREADY EXISTS
    101         000020                  F2.NEG==20                      ; NEGATIVE EXPRESSION
    102         000040                  F2.SUB==40                      ; NEGATIVE MODIFICATION
    103
    104                                 ;
    105                                 ; BINARY BUFFER INFORMATION
    106                                 ;
    107 000004                          $BIN::  .BLKW   1               ; STARTING ADDRESS
    108 000006                          $PTR::  .BLKW   1               ; CURRENT POINTER
    109 000010                          $LEFT:: .BLKW   1               ; BYTES REMAINING
    110
    111                                 ;
    112                                 ; LINE TABLE INFORMATION
    113                                 ;
    114 000012                          $SIZE:: .BLKW   1               ; NUMBER OF BYTES NEEDED
    115 000014                          $SADDR::.BLKW   1               ; SAVED REAL MEMORY ADDRESS
    116 000016                          $TADDR::.BLKW   1               ; DYNAMIC POINTER TO SCOM
    117 000020                          $ADDR:: .BLKW   1               ; ADDRESS OF BUFFER IN VRES
    118 000022                          $MXADD::.BLKW   1               ; ADDRESS OF MUX TABLES
    119 000024                          $MXSIZ::.BLKW   1               ; SIZE OF ALL MUX TABLES (.MXPTB OPERATOR)
    120 000026                          $LINK:: .BLKW   1               ; LINK VALUE
```

```
                                                .SBTTL  $VFDLC - VERIFY DLC

                                        ;+
                                        ; $VFDLC - VERIFY DLC ACTION ROUTINE
                                        ;
                                        ; INPUTS:
                                        ;       NONE
                                        ;
                                        ; OUTPUTS:
                                        ;       ERROR UNLESS DLC BEING LOADED
                                        ;-
001364                                  $VFDLC::
001364  032767  000000G 000000G                 BIT     #FL.DLC,$FLAGS  ; IS THIS A DLC?
001372  001003                                  BNE     10$             ; IF NE, YES
001374                                          EMSG$   79              ; DLC OPERATOR IN DDM/LLC TEMPLATE
001402                                  10$:    RETURN
```

```
     65
     66                                          .SBTTL  LOCAL DATA
     67                                  ;***
     68                                  ; LOCAL DATA
     69                                  ;***
     70
     71 000000                                   .PSECT  DATA,D
     72
     73                                          .NLIST  BEX
     74
     75                                  ;
     76                                  ; TEMPLATE EXTENSION
     77                                  ;
     78 000000     104     101     124   TMPEXT: .ASCIZ  "DAT"
```

```
    113                                          .IFF
    114                                          MOV     NUM,R1                  ; GET NUMBER TO CONVERT
    115                                          .ENDC
    116                                          .ENDC
    117                                          .ENDC
    118                                  MOV     #CON,R2                 ; GET CONVERSION FLAG
    119                                  .IFF
    120                                          .IF DIF <NUM>,<R1>
    121                                          MOV     NUM,R1                  ; GET NUMBER TO CONVERT
    122                                          .ENDC
    123                                  MOV     #OW,R2                  ; ELSE, CONVERT TO OCTAL WORD
    124                                  .ENDC
    125                                  CALL    CVBTO                   ; CONVERT NUMBER
    126                                  MOV     R1,MADDR                ; GET ADDRESS OF CONVERTED NUMBER
    127                                  MOV     R0,MLGTH
    128                                  MOV     #CTL,MCTL
    129                                  DIR$    #MSG
    130                                  .ENDM
    131
    132                                  .MACRO  QREAD PRMPT,?A
    133                                  .PSECT  $$$VDT,D
    134                                  $$$=.
    135                                  .ASCIZ  <15><12>%PRMPT>%
    136                                  .PSECT
    137                                  MOV     #$$$,R0                 ; GET ADDRESS OF PROMPT STRING
    138                                  CALL    PROMPT                  ; PROMPT USER FOR INPUT
    139                                  .ENDM   QREAD
    140
    141                                  .MACRO  GETNUM PROMPT,RETURN,A,?B
    142                                  .ENABL  LSB
    143                          B:      QREAD   <PROMPT>
    144                                  BCS     A
    145                                  CALL    $COTB                   ; CONVERT OCTAL TO BINARY
    146                                  TST     R2                      ; AT END OF BUFFER ?
    147                                  BNE     B                       ; IF NE, NO TRY AGAIN
    148                                  .IF DIF <RETURN>,<R1>
    149                                  MOV     R1,RETURN
    150                                  .ENDC
    151                                  .DSABL  LSB
    152                                  .ENDM   GETNUM
    153
    154                          ;
    155                          ; MACRO TO FORCE PSECT 'TEXT' TO END ON A WORD BOUNDARY
    156                          ;
    157                                  .MACRO  ENTEXT
    158                                  .PSECT  $$$VDT,D
    159                                  .EVEN
    160                                  .PSECT
    161                                  .ENDM   ENTEXT
    162
    163                          ;
    164                          ; SST TABLE IS USED IN CASE THERE IS A VTST MODULE AND NO ODT, I.E.
    165                          ; SO ONE MAY REQUEST ''OD'' (WITH NO ODT) AND NOT CRASH.
    166                          ;
    167 000030  000000           SSTTBL: .WORD   0
    168 000032  000000                   .WORD   0
    169 000034  001020'                  .WORD   BPT
```

```
**FILE**ID**VUMR
```

```
VV      VV   UU      UU   MM      MM   RRRRRRRR
VV      VV   UU      UU   MM      MM   RRRRRRRR
VV      VV   UU      UU   MMMM  MMMM   RR      RR
VV      VV   UU      UU   MMMM  MMMM   RR      RR
VV      VV   UU      UU   MM  MM  MM   RR      RR
VV      VV   UU      UU   MM  MM  MM   RR      RR
VV      VV   UU      UU   MM      MM   RRRRRRRR
VV      VV   UU      UU   MM      MM   RRRRRRRR
VV      VV   UU      UU   MM      MM   RR  RR
VV      VV   UU      UU   MM      MM   RR  RR
  VV  VV     UU      UU   MM      MM   RR    RR         ....
  VV  VV     UU      UU   MM      MM   RR    RR         ....
    VV       UUUUUUUUUU   MM      MM   RR      RR       ....
    VV       UUUUUUUUUU   MM      MM   RR      RR       ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
	.TITLE	VUNB - VNP UNLOAD BLOCK SCANNING ROUTINES
	.IDENT	/V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;	VNP - NTPOOL SCANNING ROUTINES
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00	27-FEB-78
;	VERSION 2.0 RELEASE
;
; 2.00	14-DEC-79
;	DECNET-11M/S V3.0
;	DECNET-11M-PLUS V1.0
;
; 3.00	16-APR-82
;	DECNET-11M V3.1
;	DECNET-11M-PLUS V1.1
;
; 4.00	07-NOV-83
;	DECNET-11M V4.0
;	DECNET-11M-PLUS V2.0
;
; 5.00	22-JUL-85
;	DECnet-11M/S V4.2
;	DECnet-11M-Plus V3.0
;	DECnet-Micro/RSX V1.0
;
;



```

```

                                                .SBTTL  DEVEC - DEALLOCATE VECTOR BLOCK
                                        ;+
                                        ; DEVEC - DEALLOCATE VECTOR BLOCK
                                        ;
                                        ; INPUTS:
                                        ;       R4 - PDV ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       VECTOR BLOCK DEALLOCATED IF PRESENT
                                        ;-
000000  012701  000000C                 DEVEC:  MOV     #$NTLHB+.CXVEC,R1 ; GET VECTOR BLOCK ADDRESS
000004  011100                                  MOV     (R1),R0         ; GET FIRST BLOCK
000006  001450                                  BEQ     20$             ; IF NONE - BRANCH

000010                                  5$:     CEACC$  R0,R2           ; MAP TO BLOCK
000020                                          GETAD   R2,#V.LEN       ; READ BLOCK

000036  032767  000001  000002G                 BIT     #VF.LIB,$BFR+V.FLG ; LIBRARY VECTOR BLOCK ?
000044  001006                                  BNE     7$              ; IF YES - BRANCH

000046  010446                                  MOV     R4,-(SP)        ; COPY INTERNAL PDV ADDRESS
000050  066716  000000G                         ADD     $CETDF,(SP)     ; CALCULATE ACTUAL PDV ADDRESS
000054  026726  000004G                         CMP     $BFR+V.PDV,(SP)+ ; IS IT OUR VECTOR BLOCK ?
000060  001405                                  BEQ     10$             ; IF YES - BRANCH

000062  010001                          7$:     MOV     R0,R1           ; CURRENT BLOCK BECOMES PREVIOUS BLOCK
000064  016700  000000G                         MOV     $BFR,R0         ; GET NEXT BLOCK
000070  001347                                  BNE     5$              ; CONTINUE

000072                                          RETURN

000074  022701  000000C                 10$:    CMP     #$NTLHB+.CXVEC,R1 ; IS PREVIOUS BLOCK THE LISTHEAD ?
000100  001003                                  BNE     15$             ; IF NOT - BRANCH
000102  016711  000000G                         MOV     $BFR,(R1)       ; IF YES - UPDATE LISTHEAD
000106  000404                                  BR      18$             ; CONTINUE

000110  010146                          15$:    MOV     R1,-(SP)        ; COPY UNMAPPED ADDRESS OF PREVIOUS BLOCK
000112  010046                                  MOV     R0,-(SP)        ; COPY UNMAPPED ADDRESS OF CURRENT BLOCK
000114                                          CALL    $XLINK          ; UNLINK VECTOR BLOCK

000120  012701  000012                  18$:    MOV     #V.LEN,R1       ; DEALLOCATE VECTOR BLOCK
000124                                          CALL    $DEPOL          ; ...

000130                                  20$:    RETURN

```

```
    744 001740   010467  000202'                MOV     R4,..PDV         ; SAVE THE PDV ADDRESS
    745 001744   016467  000000G 000000G        MOV     Z.DSP(R4),$BIAS  ; GET THE PDV BIAS
    746
    747 001752                                  CALL    VECT             ; RELEASE VECTOR(S)
    748 001756   016500  000000G                MOV     L.KRBA(R5),R0    ; GET THE KRB ADDRESS
    749 001762                                  CALL    XKRB             ; DEALLOCATE THE KRB
    750
    751 001766                                  CALL    XLT              ; DE-ALLOCATE LINE TABLE
    752 001772                                  CALL    ZDDS             ; ZERO L.DDS IN SLT(S)
    753 001776   016704  000202'                MOV     ..PDV,R4         ; DDM PDV ADDRESS
    754 002002   016467  000000G 000000G        MOV     Z.NAM(R4),$NAME  ; SETUP DDM DEVICE NAME
    755 002010   005767  000122'         5$:    TST     DDMAC            ; ANY OTHER DDM ACTIVITY?
    756 002014   001006                         BNE     10$              ; IF NE, YES
    757 002016   132764  000000G 000000G        BITB    #ZF.MFL,Z.FLG(R4) ; IS DDM PROCESS MARKED FOR LOAD?
    758 002024   001002                         BNE     10$              ; IF NE, YES - DON'T REMOVE
    759 002026                                  CALL    XPRO             ; REMOVE PROCESS FROM CORE
    760 002032   126565  000000G 000000G 10$:   CMPB    L.DDM(R5),L.DLC(R5) ; COMBINATION DDM/DLC?
    761 002040   001434                         BEQ     30$              ; IF EQ, YES - ALL DONE
    762 002042   016503  000000G         20$:   MOV     L.DLS(R5),R3     ; GET DLC LINE TABLE ADDRESS
    763 002046   001437                         BEQ     111$             ; IF ZERO, VERY BAD
    764 002050   116504  000000G                MOVB    L.DLC(R5),R4     ; GET DLC PDV INDEX
    765 002054                                  CALL    GETPDV           ; CONVERT TO PDV ADDRESS
    766 002060   010467  000202'                MOV     R4,..PDV         ; AND SAVE
    767
    768 002064   116504  000000G                MOVB    L.DLC(R5),R4     ; GET DLC PDV INDEX
    769 002070                                  CALL    XLT16            ; DE-ALLOCATE DLC LINE TABLE
    770 002074   005065  000000G                CLR     L.DLS(R5)        ; ZERO LINE TABLE ADDRESS
    771 002100   005065  000000G                CLR     L.DLM(R5)        ; AND LINE TABLE BIAS
    772 002104   005767  000120'                TST     DLCAC            ; ANY OTHER DLC ACTIVITY?
    773 002110   001010                         BNE     30$              ; IF NE, YES
    774 002112   016704  000202'                MOV     ..PDV,R4         ; DLC PDV ADDRESS
    775 002116   132764  000000G 000000G        BITB    #ZF.MFL,Z.FLG(R4) ; IS DLC PROCESS MARKED FOR LOAD?
    776 002124   001002                         BNE     30$              ; IF NE, YES - DON'T REMOVE
    777 002126                                  CALL    XDLC             ; REMOVE DLC FROM CORE
    778 002132   042765  000000G 000000G 30$:   BIC     #LF.UNL,L.FLG(R5) ; CLEAR MARKED FOR UNLOAD FLAG
    779 002140   000241                         CLC                      ; RETURN C-BIT CLEAR
    780
    781 002142                                  RETURN
    782
    783                                  ;
    784                                  ; ERROR CONDITIONS
    785                                  ;
    786 002144                           101$:  CRASH                    ; KRB ADDRESS IS ZERO !
    787 002146                           111$:  CRASH                    ; DLC LINE TABLE ADDRESS IS ZERO !
    788
    789
    790                                  ;
    791                                  ; CONVERT PDV INDEX TO PDV ADDRESS
    792                                  ;
    793 002150   066704  000000C         GETPDV: ADD    $PDVTA+AB,R4     ; ADD PDV TABLE ADDRESS
    794 002154   016404  000000G                MOV     AB(R4),R4        ; GET TABLE ENTRY
    795 002160                           10$:   RETURN
    796
    797
    798                                  ;
    799                                  ; DE-ALLOCATE KRB
    800                                  ;
```

```
                                                .SBTTL  $CCEX - CLEAR SYS
                                        ;+
                                        ; $CCEX - CLEAR SYSTEM
                                        ;
                                        ; THIS ROUTINE IS CALLED TO REMOVE ALL COMM EXEC RELATED PIECES FROM CORE.
                                        ;
                                        ; INPUTS:
                                        ;       NONE
                                        ;
                                        ; OUTPUTS:
                                        ;       EVERYTHING IS REMOVED FROM CORE
                                        ;-

003666                                  $CCEX::
                                                .MCALL  VNPBPT
003666                                          VNPBPT  CCEX
003666  032767  000000G 000000G                 BIT     #FE.CEX,$FMASK    ; IS COMM EXEC LOADED ?
003674  001002                                  BNE     1$                ; IF YES - BRANCH
003676  000241                                  CLC                       ; IF NO - RETURN SUCCESS
003700                                          RETURN

003702  012700  000124'                 1$:     MOV     #NTIR,R0          ; STORE NTINIT'S RAD50 TCB NAME
003706                                          CALL    TSKSCH            ; GET TCB ADDRESS
003712  016705  000000G                         MOV     $RADDR,R5         ; ...
003716                                          GETRV   R5,.TLGTH         ; READ NTINIT TCB
003734  016700  000012G                         MOV     T.RCVL+$BFR,R0    ; GET PACKET ADDRESS
003740  001424                                  BEQ     5$                ; IF NONE - BRANCH
003742  005067  000012G                         CLR     T.RCVL+$BFR       ; REMOVE ENTRY FROM RECIEVE QUEUE
003746  010504                                  MOV     R5,R4             ; COPY TCB ADDRESS
003750  062704  000012                          ADD     #T.RCVL,R4        ; POINT TO RECIEVE QUEUE
003754  010467  000014G                         MOV     R4,T.RCVL+2+$BFR ; POINT QUEUE TO ITSELF
003760                                          PUTRC   R5,.TLGTH         ; WRITE TCB TO IMAGE
003776  012701  000000G                         MOV     #DSIZE,R1         ; STORE SEND DATA PCKT LENGTH

                                                .IF DF  R$$MPL                                                  ;[MP01]
                                                BIT     #DF.POL,$DFLAG    ; SECONDARY POOL SUPPORTED ?          ;[MP01]
                                                BEQ     2$                ; IF NO - BRANCH                      ;[MP01]
                                                CALL    .DESPK            ; DEALLOCATE PACKET                   ;[MP01]
                                                BR      4$                ; CONTINUE                            ;[MP01]

                                                .IFTF                                                           ;[MP01]
004002                                  2$:     CALL    $DEA16            ; DEALLOCATE PACKET
                                                .ENDC                                                           ;[MP01]

004006                                  4$:     CALL    CLQREM            ; REMOVE NTI AND TIMER ENTRIES FROM CLOCK QUEUE
004012                                  5$:
004012  016704  000000G                         MOV     $SLTMA,R4         ; GET SLT MAPPING TABLE ADDRESS
004016  016746  000000G                         MOV     $SLTNM,-(SP)      ; AND COUNT OF SLT'S
004022  001415                                  BEQ     30$               ; IF ZERO, NO WORK HERE
004024  012405                          10$:    MOV     (R4)+,R5          ; GET NEXT SLT
004026  032765  000000G 000000G                 BIT     #LF.RDY,L.FLG(R5) ; IS THE LINE READY?
004034  001406                                  BEQ     20$               ; IF EQ, NO
004036  010567  000000G                         MOV     R5,$SLTA          ; SAVE SLT ADDRESS
004042  010446                                  MOV     R4,-(SP)          ; SAVE POINTER INTO SLT MAPPING TABLE
004044                                          CALL    XLIN              ; UNLOAD THE LINE
004050  012604                                  MOV     (SP)+,R4          ; RESTORE MAPPING INTO SLT TABLE
004052  005316                          20$:    DEC     (SP)              ; LOOP <$SLTNM> TIMES
```

```
                                                               K  1
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55       PAGE 3
SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL  VALUE              REFERENCES

LO.SNM    000035           #5-63
LO.TPA    000020           #5-63
LO.TRB    000034           #5-63
LO.TRI    000030           #5-63
LO.UNT    000033           #5-63
LO.VCT    000014           #5-63
LO.VER    000042           #5-63
LR.ACC    000020           #5-63
LR.ACO    000052           #5-63
LR.ADD    000012           #5-63
LR.ALI    000002           #5-63
LR.BLK    000010           #5-63
LR.CTL    000004           #5-63
LR.DES    000010           #5-63
LR.HLP    000002           #5-63
LR.LIN    000002           #5-63         5-63      5-63      5-63      5-63      5-63      5-63
LR.NAM    000012           #5-63
LR.PAS    000041           #5-63
LR.PRO    000002           #5-63
LR.STS    000000           #5-63         26-972
LR.TRI    000006           #5-63
LR.TYP    000002           #5-63
LR.UCB    000016           #5-63
LR.UIC    000020           #5-63
LR.UNT    000005           #5-63
LS.ALI  = 000016           #5-63
LS.CEX  = 000001           #5-63
LS.CIR  = 000033           #5-63
LS.CXO  = 010000           #5-63         26-972
LS.CX5  = 000400           #5-63
LS.ECH  = 001000           #5-63         7-192
LS.FDX  = 004000           #5-63         7-190
LS.HDX  = 010000           #5-63         7-191
LS.HLP  = 000012           #5-63
LS.LIN  = 000003           #5-63
LS.LMC  = 000007           #5-63
LS.LOG  = 000020           #5-63
LS.MOD  = 000036           #5-63
LS.NOD  = 000014           #5-63
LS.NTI  = 000013           #5-63
LS.OBJ  = 000022           #5-63
LS.OPT  = 000400           #5-63         16-503    20-669
LS.PRO  = 000002           #5-63
LS.TOP  = 002000           #5-63         7-193
LS.TST  = 000010           #5-63
LS.UNF  = 020000           #5-63
LS.XIT  = 000011           #5-63
LX.ALI  = 000017           #5-63
LX.CEX  = 000004           #5-63
LX.CIR  = 000034           #5-63
LX.LIN  = 000006           #5-63
LX.LOG  = 000021           #5-63
```

```
                                                               L  1
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55       PAGE 4
```

```
                                                                           K  2
VSET - VNP SETUP CSR AND MICROC MACRO V05.03b Saturday 29-Jun-85 02:59  Page 6-1
Symbol table

A$$CHK= 000000             F.NRBD= ****** GX          L$$ASG= 000000             N$$MLL= 000001             T$$KMG= 000000
A$$CPS= 000000             F.RSIZ= ****** GX          L$$DRV= 000000             N$$MOV= 000010             T$$MIN= 000000
A$$PRI= 000000             F.URBD= ****** GX          L$$P11= 000001             N$$NCT= 000001             V$$CTR= 001000
A$$TRP= 000000             G$$TPP= 000000             L$$11R= 000000             N$$PEM= 000001             X$$DBT= 000000
C$$CKP= 000000             G$$TSS= 000000             M$$CRB= 000124             P$$P45= 000000             $BFR   = ****** GX
C$$ORE= 000400             G$$TTK= 000000             M$$CRX= 000000             P$$WRD= 000000             $CETDF= ****** GX
C$$RSH= 177564             G$$WRD= 000000             M$$FCS= 000000             Q$$OPT= 000010             $CSRLD  000000RG
D$$BUG= 177514             I$$RAR= 000000             M$$MGE= 000000             R$$DER= 000000             $NTLHB= ****** GX
D$$ISK= 000000             I$$RDN= 000000             M$$NET= 000000             R$$K11= 000001             $PUTAD= ****** GX
D$$L11= 000001             K$$CNT= 177546             M$$OVR= 000000             R$$SND= 000000             $SLTA  = ****** GX
D$$YNC= 000000             K$$CSR= 177546             N$$ACC= 000001             R$$11M= 000000             $UNBLK= ****** GX
D$$YNM= 000000             K$$LDC= 000000             N$$BUF= 000001             SYSFDB= ****** GX          $VUNB  = ****** GX
E$$XPR= 000000             K$$TPS= 000074             N$$LDV= 000001             S$$WRG= 000000             $$CSR  = ****** GX
F$$LVL= 000001             LD$LP  = 000000            N$$MCP= 000001             S$$YSZ= 007600             .CXCSR= ****** GX


. ABS.  000000    000    (RW,I,GBL,ABS,OVR)
        000152    001    (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 9350 Words  ( 37 Pages)
Size of core pool: 14440 Words  ( 55 Page^)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.33
SY:VSET.V2,[132,134]VSET/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VSET
```

```
                                                                  L  2
VSET        CREATED BY  MACRO  ON 29-JUN-85 AT 02:59      PAGE 1
```

```
    349                          .SBTTL  SET COMMAND
    350
    351                          ;       Keyword         Next      Action      Mask (BIS)     Flag to
    352                          ;       Type, or        State     Routine       Value         Mask
    353                          ;       Substate
    354
    355                  ;
    356                  ; SET COMMAND
    357                  ;
    358 000000                   STATE$  SET,&P$SET
    359 000000                    TRAN$  !IDALI,         SEAL,        ,          LS.ALI,        RQBST
    360
    361 000000                    TRAN$  !IDCIR,         SECI,        ,          LS.CIR,        RQBST
    362 000000                    TRAN$  !IDKNCI,        SECI,                   LS.CIR,        RQBST
    363 000000                    TRAN$  !IDLIN,         SELI,     $XMISS,       LS.LIN,        RQBST
    364 000000                    TRAN$  !IDKNLI,        SELI,        ,          LS.LIN,        RQBST
    365
    366 000000                    TRAN$  !IDLOG,         SELO,        ,          LS.LOG,        RQBST
    367 000000                    TRAN$  !IDKNLO,        SELO,        ,          LS.LOG,        RQBST
    368 000000                    TRAN$  !IDMOD,         SEMO,        ,          LS.MOD,        RQBST
    369 000000                    TRAN$  !IDNOD,         SENO,        ,          LS.NOD,        RQBST
    370 000000                    TRAN$  !IDOBJ,         SEOB,        ,          LS.OBJ,        RQBST
    371 000000                    TRAN$  !IDPRO,         SEPR,        ,          LS.PRO,        RQBST
    372 000000                    TRAN$  !IDSYS,         SESY,        ,          LS.NTI,        RQBST
    373                  ;
    374                  ; SET ALIAS
    375                  ;
    376 000000                   STATE$  SEAL,&P$SEAL
    377 000000                    TRAN$  !OPSCO,         SEAL1,       ,          OP$SCO,        $OPTON
    378 000000                    TRAN$  !OPDES,         SEAL2
    379                  ;
    380 000000                   STATE$  SEAL1,&P$SEAL
    381 000000                    TRAN$  !OPDES,         DONE
    382                  ;
    383 000000                   STATE$  SEAL2
    384 000000                    TRAN$  !OPSCO,         DONE,        ,          OP$SCO,        $OPTON
    385 000000                    TRAN$  $LAMDA,         DONE
    386                  ;
    387                  ; SET CIRCUIT
    388                  ;
    389
    390 000000                   STATE$  SECI,&P$SECI
    391 000000                    TRAN$   $LAMDA,        SECIOP
    392 000000                   STATE$  SECIOP
    393 000000                    TRAN$  !OPCOS,         SECIOP,      ,          OP$COS,        $OPTON
    394 000000                    TRAN$  !OPHTM,         SECIOP,      ,          OP$HTM,        $OPTON
    395 000000                    TRAN$  !OPLTM,         SECIOP,      ,          OP$LTM,        $OPTON
    396 000000                    TRAN$  !OPMACT,        SECIOP,      ,          OP$MAC,        $OPTON
    397 000000                    TRAN$  !OPOWN,         SECIOP,      ,          OP$OWN,        $OPTON
    398 000000                    TRAN$  !OPLNST,        SECIOP,      ,          OP$STA,        $OPTON
    399 000000                    TRAN$  !OPTRI,         SECIOP,      ,          OP$TRI,        $OPTON
    400 000000                    TRAN$  !OPSER,         SECIOP,      ,          OP$SER,        $OPTON
    401 000000                    TRAN$  $LAMDA,         DONE
    402
    403                  ;
    404                  ; SET LINE
    405                  ;
```

```
000000                                    TRAN$   $LAMDA,        OPDST           ; Must be DESTINATION ...       ;[TD001]
                                                                                                                 ;[TD001]
000000                                   STATE$  OPAD2                                                           ;[TD001]
000000                                    TRAN$   &K$NETS,       $EXIT,          ,       OP$KNT,         $OPTON  ;[TD001]
000000                                    TRAN$   $LAMDA,        OPDST1          ; Must be KNOWN DESTINATIONS    ;[TD001]
                                                                                                                 ;[TD001]
000000                                   STATE$  OPANT,&P$NET                    ; Parse network name            ;[TD001]
000000                                    TRAN$   $RAD50,        $EXIT,      $ANT,       OP$NET,         $OPTON  ;[TD001]
                                                                                                                 ;[TD001]
                                 ;                                                                               ;[TD001]
                                 ;    DESTINATION for SHO X2n-SERVER                                             ;[TD001]
                                 ;                                                                               ;[TD001]
                                                                                                                 ;**-3
000000                                   STATE$  OPDST
000000                                    TRAN$   &K$KNO,        OPDST1
000000                                    TRAN$   &K$DES
000000                                   STATE$  ,&P$DST                                                         ;[TD001]
000000                                    TRAN$   $RAD50,        $EXIT,      $ADS,       OP$DST,         $OPTON  ;[TD001]
000000                                   STATE$  OPDST1                                                          ;**-2
000000                                    TRAN$   &K$DESS,       $EXIT,      ,           OP$KDS,         $OPTON
                                 ;
                                 ; DTE
                                 ;
000000                                   STATE$  OPPDT
000000                                    TRAN$   &K$KNO,        OPPDT1
000000                                    TRAN$   &K$DTE
000000                                   STATE$  ,&P$DTE
000000                                    TRAN$   $STRNG,        $EXIT,      $PDT,       OP$DTE,         $OPTON
000000                                   STATE$  OPPDT1
000000                                    TRAN$   &K$DTES,       $EXIT,      ,           OP$KDT,         $OPTON
                                 ;
                                 ; DUPLEX
                                 ;
000000                                   STATE$  OPDUP
000000                                    TRAN$  &K$DUP
000000                                   STATE$  ,&P$DUP
000000                                    TRAN$  &K$FUL,         $EXIT,      ,           LS.FDX,         RQBST
000000                                    TRAN$  &K$HAL,         $EXIT,      ,           LS.HDX,         RQBST
                                 ;
                                 ; EVENT
                                 ;
000000                                   STATE$  OPEVE
000000                                    TRAN$  &K$KNO,         OPEVE1
000000                                    TRAN$  &K$EVE
000000                                   STATE$  ,&P$EVE
000000                                    TRAN$  !EVEID,         $EXIT
000000                                   STATE$  OPEVE1
000000                                    TRAN$  &K$EVE,         $EXIT,      ,           OP$KEV,         $OPTON
                                 ;
                                 ; GROUP
                                 ;
000000                                   STATE$  OPGRO
000000                                    TRAN$   &K$KNO,        OPGRO1
000000                                    TRAN$   &K$GRP
000000                                   STATE$  ,&P$GRP
000000                                    TRAN$   $RAD50,        $EXIT,      $GRO,       OP$GRP,         $OPTON
```

```
    1607                                                 .SBTTL  STMAT3, STMAT5 - SET MINIMUM CHARACTER MATCH
    1608                                         ;+
    1609                                         ;
    1610                                         ; ***  STMAT3, STMAT5 - SET MINIMUM NUMBER OF CHARACTERS TO MATCH FOR
    1611                                         ;                        KEYWORD
    1612                                         ;
    1613                                         ; INPUTS:    NONE
    1614                                         ;
    1615                                         ; OUTPUTS:
    1616                                         ;
    1617                                         ;    .PFLAG - HIGH BYTE = MINIMUM NUMBER OF CHARACTERS TO MATCH
    1618                                         ;-
    1619 000020                                  STMAT1:
    1620 000020  112767  000001  000001G                 MOVB    #1,.PFLAG+1             ;SET 1 CHARACTER MINIMUM
    1621 000026                                          RETURN
    1622                                         ;
    1623 000030                                  STMAT3:
    1624 000030  112767  000003  000001G                 MOVB    #3,.PFLAG+1             ;SET 3 CHARACTER MINIMUM
    1625 000036                                          RETURN
    1626                                         ;
```

```
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL  VALUE              REFERENCES

TY$STA  = 002000           #5-62
TY$SUM  = 000000           #5-62
VF.ADD  = ******   G        5-86
VF.ECH  = ******   G        5-86
VF.FIX  = ******   G        5-86
VF.ON   = ******   G        5-86
VF.PEM  = ******   G        5-86       19-1642
VF.PSI  = ******   G        5-87
VF.SED  = ******   G        5-87
WC.CNT  = 000400           #5-62
WC.EVE  = 004000           #5-62
WC.TRI  = 002000           #5-62
WC.UNT  = 001000           #5-62
$ADS    = ******   G        5-90
$ALPHA  = 000022           #6-108
$ANT    = ******   G        5-90
$ANY    = 000020           #6-108
$ARAD1  = ******   G        5-100
$ARAD2  = ******   G        5-100
$BLANK  = 000006           #6-108
$CKLOG  = ******   G        5-90
$CMAND  = ******   G        5-84
$COP    = ******   G        5-90
$COST   = ******   G        5-90
$CSR    = ******   G        5-90
$CTL    = ******   G        5-90
$DACC   = ******   G        5-91
$DIGIT  = 000024           #6-108
$DNUMB  = 000014           #6-108
$DPAS   = ******   G        5-91
$DUSE   = ******   G        5-91
$EOS    = 000012           #6-108
$EXIT   = 000000           #6-108
$EXNOD  = ******   G        5-91       19-1643
$FAIL   = 177777           #6-108
$FLAGS  = ******   G        5-84
$GOFF   = 000001           #5-66      6-108
$GPRM   = 000001           #5-65      6-108
$GRO    = ******   G        5-91
$HLPAC  = ******   G        5-91
$HTM    = ******   G        5-92
$IFALI  = ******   G        5-92
$IFLOG  = ******   G        5-92
$IFNNA  = ******   G        5-92
$IFXAC  = ******   G        5-92
$IFXPR  = ******   G        5-92
$IFX5S  = ******   G        5-93
$IFX9S  = ******   G        5-93
$INA    = ******   G        5-100
$INC    = ******   G        5-100
$LAMDA  = 000000           #6-108
$LNAME  = ******   G        5-93
```

```
    151 000064                      .CXLEN::.BLKW   1               ; CETAB ALLOCATION LENGTH
    152 000066                      .SBPCB::.BLKW   1               ; SERVER BASE BUFFER POOL PCB ADDRESS
    153 000000                              .PSECT
    154
    155                             ;
    156                             ; COMM EXEC ADDRESSING BIAS
    157                             ;
    158             000000          AB==0
    159
    160                             ;
    161                             ; LIBRARY BLOCK OFFSETS
    162                             ;
    163 000070                              .ASECT
    164             000000                  .=0
    165 000000                              .BLKW   3               ; LIBRARY NAME IN RAD50
    166 000006                      LB.ADR::.BLKW   1               ; ADDRESS/BIAS
    167 000010                      LB.SIZ::.BLKW   1               ; SIZE
    168 000012                      LB.USE::.BLKW   1               ; USE COUNT
    169             000013          LB.APR==LB.USE+1                ; STARTING APR (MAPPED ONLY)
    170 000000                              .PSECT
    171
    172             000001                  .END
    121 000030                  $OFF::  .BLKW   1               ; OFFSET VALUE
    122 000032                  $MXLEN::.BLKW   1               ; MUX TABLE LENGTH (STORED AS NEGATIVE VALUE)
    123 000034                  $NALL:: .BLKB   1               ; NUMBER OF ALLOCATIONS
    124 000035                  $NLIB:: .BLKB   1               ; NUMBER OF LIBRARY REFERENCES
    125 000036                  $SZPTR::.BLKW   1               ; POINTER INTO SIZE TABLE
    126 000040                          .BLKW   6               ; $SIZE SAVED EACH TIME A NEW LEVEL IS ENTERED
    127 000054                  $UBIAS::.BLKW   1               ; UNLOAD BLOCK BIAS
    128 000056                  $UDA::  .BLKW   1               ; UNLOAD DATA VIRTUAL ADDRESS
    129 000060                  $LDA::  .BLKW   1               ; LIBRARY DATA VIRTUAL ADDRESS
    130
    131                         ;
    132                         ; NETLDR ACCOUNTING BLOCK IMAGE
    133                         ;
    134 000062                  $NTLUB::.BLKW   32.             ; IMAGE OF CURRENT UNLOAD DATA BLOCK
    135 000162                  $NTLLB::.BLKW   32.             ; IMAGE OF CURRENT LIBRARY DATA BLOCK
    136
    137                         ;
    138                         ; DDM VECTOR INFORMATION
    139                         ;
    140 000262                  $VECT1::.BLKW   1               ; DISPLACEMENT OF ".INT1" FIELD
    141 000264                  $VECT2::.BLKW   1               ; DITTO ".INT2"
    142 000266                  $VECT3::.BLKW   1               ; DITTO ".INT3"
    143 000270                  $NVECT::.BLKW   1               ; NUMBER OF VECTORS
    144 000272                  $ICBAD::.BLKW   1               ; ADDRESS OF ICB
    145 000274                  $ICBSZ::.BLKW   1               ; AND ITS LENGTH
    146
    147                         ;
    148                         ; TPARS ACTION ROUTINES INFORMATION
    149                         ;
    150 000276                  $VALUE::.BLKW   1               ; EXPRESSION VALUE
    151 000300                  $DIFF:: .BLKW   1               ; MODIFICATION VALUE
    152 000302                  $FILEN::.BLKB   10.             ; ASCIZ FILE NAME
    153 000314                  $CVAL:: .BLKW   1               ; ".CORE" OPERAND 1 VALUE
    154 000316                  $CAPR:: .BLKW   1               ; ".CORE" OPERAND 2 APR NUMBER
    155 000320                  $MPLD:: .BLKW   1               ; ".MPLHD" DISPLACEMENT
    156 000322                  $LFLD:: .BLKW   1               ; ".LFLHD" DISPLACEMENT
    157 000324                  $BFALL::.BLKW   1               ; ".BIN" ALLOCATION SIZE IN 32. WORD BLOCKS
    158 000326                  $BFLBN::.BLKW   2               ; ".BIN" LOGICAL BLOCK NUMBER
    159 000332                  $BFPTR::.BLKW   1               ; ".BIN" BINARY BUFFER POINTER
    160 000334                  $BFXFR::.BLKW   1               ; ".BIN" TRANSFER SIZE IN BYTES
    161 000336                  $BFBAS::.BLKW   1               ; BINARY/LIBRARY FILE TKB BASE ADDRESS
    162 000340                  $FNR50::.BLKW   3               ; FILE NAME IN RAD50
    163 000346                  $LBADR::.BLKW   1               ; ".LIBR" DESCRIPTOR BLOCK ADDRESS
    164 000350                  $LBIAS::.BLKW   1               ; ".LIBR" DESCRIPTOR BLOCK BIAS
    165
    166                         ;
    167                         ; TEMPLATE BUFFER AND BUFFER SIZE
    168                         ;
    169 000352                  TMPBF:  .BLKB   132.
    170 000556                  TMPSZ:  .BLKW   1
```

```
                                        .SBTTL  $VFSPC - VERIFY SPECIAL OPERATOR
                                        .SBTTL  $VFLV1 - VERIFY LEVEL 1
                                ;+
                                ; $VFSPC - VERIFY SPECIAL OPERATOR
                                ; $VFLV1 - VERIFY PRIMARY TEMPLATE (LEVEL 1)
                                ;
                                ; INPUTS:
                                ;       R0=$FLAG1 BIT CODE FOR THE OPERATOR ($VFSPC ONLY)
                                ;
                                ; OUTPUTS:
                                ;       OPERATOR BIT SET IN $FLAG1
                                ;       ERROR IF OPERATOR HAS ALREADY BEEN SEEN OR IF NOT AT LEVEL 1
                                ;-

                                        .ENABL  LSB

001404                          $VFSPC::
001404  030067  000000'                 BIT     R0,$FLAG1       ; IT THIS THE FIRST TIME FOR THIS GUY?
001410  001007                          BNE     101$            ; IF NE, NO
001412  050067  000000'                 BIS     R0,$FLAG1       ; INDICATE OPERATOR SEEN
001416                          $VFLV1::
001416  026727  000036' 000040'         CMP     $SZPTR,#$SZPTR+2 ; IS THIS THE TOP LEVEL?
001424  001004                          BNE     111$            ; IF NE, NO
001426                                  RETURN                  ; (C-BIT CLEAR)

                                ;
                                ; ERROR CONDITIONS
                                ;
001430                          101$:   EMSG$R  82              ; DUPLICATE OPERATOR
001436                          111$:   EMSG$R  83              ; PRIMARY TEMPLATE OPERATOR

                                        .DSABL  LSB
```

```
                                                        .SBTTL  ERROR MESSAGES
                                                ;
                                                ; ERROR MESSAGES
                                                ;
                                                        .ENABL  LC

        000004                                          NTLER$  ,8,3,$TWE,$PRV.N,$FNOUT,<Open Failure (-***.)>
        000042                                          NTLER$  ,WZ,,$TWARN,RTSPC,,<MUX update only>

                                                        .DSABL  LC

                                                ;
                                                ; ERROR MESSAGE FORMAT STRING
                                                ;
                                                        .ENABL  LC
                                                        .NLIST  BEX
        000072      000                         FMT0:   .ASCIZ  ""
        000073      052     040     124         FMT3:   .ASCIZ  "* Template File -- "
                                                        .EVEN
                                                        .DSABL  LC
                                                        .LIST   BEX

        000000                                          .PSECT
```

```
    170
```

```
                                        .TITLE  VUMR - VNP DYNAMIC UMR ALLOC/DEALLOC ROUTINES
                                        .IDENT  /V05.00/

                                ;
                                ; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ;       VNP - DYNAMIC UMR ALLOCATION/DE-ALLOCATION ROUTINES
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  27-FEB-78
                                ;       VERSION 2.0 RELEASE
                                ;
                                ; 2.00  14-DEC-79
                                ;       DECNET-11M/S V3.0
                                ;       DECNET-11M-PLUS V1.0
                                ;
                                ; 3.00  16-APR-82
                                ;       DECNET-11M V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;
```

```
;+
;       $VUNB   - SCAN NTPOOL CHAIN OF BLOCKS
;
; INPUTS:
;       R0      - ADDRESS OF LINK WORD
;       R1      - NUMBER OF WORDS/SEGMENT
;       R2      - ADDRESS OF SEGMENT WITHIN BLOCK BUFFER
;
; OUTPUTS:
;       R0      - POINTER TO $UNBLK ( MUST BE PRESERVED ACROSS CALLS
;       R1      - UNCHANGED
;       R2      - ADDRESS OF SEGMENT IN BUFFER ( $UNBLK )
;
; NOTES:
;
; R0 MUST BE INPUT AS A LISTHEAD ADDRESS IN THE HOME BLOCK ( $NTLHB ).
;       ONCE THE ROUTINE ( $VUNB ) HAS BEEN CALLED, R0 WILL BE CHANGED
;       TO POINT AT $UNBLK. THIS ADDRESS MUST BE PRESERVED ACROSS CALLS
;       TO $VUNB.
; IF WHEN CALLED R2 IS NON-ZERO IS IS ASSUMED TO BE THE SEGMENT ADDRESS
;       RETURNED TO THE CALLER BY A PREVIOUS CALL TO THIS ROUTINE, AND THE
;       ALGORITHIM WILL RETURN THE ADDRESS OF THE NEXT SEGMENT IN R2
;       EVEN IF THAT SEGMENT IS WITHIN THE NEXT BLOCK IN THE CHAIN. IF
;       NO NEXT BLOCK EXISTS, THE R2 WILL BE RETURNED AS ZERO AND THE
;       C-BIT WILL BE RETURNED SET. ALL OTHER ERRORS ARE FATAL. IF WHEN
;       CALLED R2 IS ALREADY ZERO AND THE LINK WORD IS ZERO ( I.E., (R0)
;       = 0 ), THEN A NEW BLOCK WILL BE ALLOCATED, ZEROED, REWRITTEN TO
;       THE DISK FILE, AND LINKED TO THE CHAIN. THEN R2 WILL BE RETURNED
;       POINTING TO THE SECOND WORD OF THIS NEW BLOCK. A SPECIAL CASE IS
;       ENCOUNTERED WHEN R0 IS THE ADDRESS OF THE LISTHEAD IN THE HOME
;       BLOCK ; THIS CASE IS HANDLED TRANSPARENTLY.
; THE SWTCHI AND SWTCHO MACROS ARE INVOKED SO THAT UPON EXIT THE
;       DEFAULT BUFFER IS $BFR REGARDLESS OF THE DEFAULT BUFFER ON
;       INPUT. IF THE CALLER WANTS TO REWRITE A MODIFIED BLOCK
;       THEN THE SWTCHI/O MACROS MUST BE USED TO POINT AT THE
;       BLOCK ($UNBLK) IN ORDER TO REWRITE TO THE SYSTEM IMAGE
;       FILE. THE ADDRESS AND SIZE FIELDS ARE SET UP PROPERLY
;       ASSUMING THE USER HAS NOT ISSUED ANY OTHER GETXX/PUTXX
;       MACROS SINCE CALLING THIS SUBROUTINE.
; SEGMENTS ARE SCANNED BY STARTING WITH THE SECOND WORD OF EACH
;       BLOCK AND INCREMENTING R1 WORDS TO THE BEGINNING OF THE
;       NEXT SEGEMENT. SEGEMNETS CANNOT CROSS BLOCK BOUNDARIES.
; THIS ROUTINE ASSUMES THAT A ROOT SEGMENT BLOCK BUFFER AND DESCRIPTOR
;       ALL DEDICATED TO IT'S USE:
;               $UNBAD  - UNLOAD BLOCK VIRTUAL ADDRESS
;               $UNBLK  - 32. WORD BLOCK BUFFER
;               $UNEND  - FIRST WORD NOT IN BLOCK BUFFER
;
;-
```

```


                                                .SBTTL  $XPRO - UNSUCCESSFUL SET PROCESS CLEANUP
                                        ;+
                                        ; $XPRO - CLEANUP AFTER AN UNSUCCESSFUL "SET PROCESS" (RSX11M ONLY)
                                        ;
                                        ; THIS ROUTINE IS CALLED IF A "SET PROCESS" FAILS EITHER IN SETTING UP THE
                                        ; LLC LINE TABLE OR IN SETTING UP THE RSX11M DCB DATA STRUCTURES. THE PROCESS
                                        ; WILL BE RELEASED FROM ITS PARTITION AND, IF IT IS AN LLC, ITS LINE TABLE WILL
                                        ; BE DE-ALLOCATED.
                                        ;
                                        ; INPUTS:
                                        ;       $PDVA=PDV ADDRESS
                                        ;       FL.PX1 OF $FLAGS SET IF PROCESS WAS TRANSFERRED INTO CORE
                                        ;
                                        ; OUTPUTS:
                                        ;       C-BIT=SET
                                        ;-

                                                .ENABL  LSB

000132                                  $XPRO::
                                                .MCALL  VNPBPT
000132                                          VNPBPT  XPRO
000132  016704  000000G                         MOV     $PDVA,R4                ; GET PDV ADDRESS
000136  032767  000000G 000000G                 BIT     #FL.PX1,$FLAGS          ; WAS PROCESS TRANSFERRED?
000144  001002                                  BNE     XPRO                    ; IF YES - BRANCH
000146                                          CALLR   10$                     ; CONTINUE

000152                                  XPRO:
                                                .IF DF  R$$MPL                                                  ;[MP01]

                                                CALL    DEVEC                   ; DEALLOCATE VECTOR BLOCK       ;[MP01]
                                                BITB    #ZF.DDM,Z.FLG(R4)       ; DDM ?                         ;[MP01]
                                                BEQ     5$                      ; IF NO - BRANCH                ; '''

                                                MOV     $NAME,CTBSYM+2          ; FORM SYMBOL NAME TO LOOK FOR "$CTXXX"
                                                MOV     #$DFUIC,R0              ; GET THE COMM EXEC UIC
                                                MOV     #CETNAM,R1              ; FORM ASCIZ FILE SPEC "[XX,5X]CETAB.STB"
                                                MOV     #STBEXT,R2              ;       ...
                                                CALL    $AZFS                   ;       ...
                                                MOV     #CTBLK,R0               ; GET SYMBOL BLOCK TO LOOK FOR
                                                CALL    $SYMBL                  ; LOOK FOR IT

                                                SWTCHI                          ; SET DEFAULT BUFFERS
                                                SWTCHO                          ;       ...
                                                MOV     $NTLHB+.CXALL,R1        ; GET START OF CETAB
                                                ADD     CTVAL,R1                ; CALCULATE FIRST CTB ADDRESS

                                                MOV     R1,$CTBAD               ; SAVE CTB ADDRESS
                                        ;
                                        ; SEE IF CTB ALREADY IN LIST
                                        ;

                                                MOV     $CTLST,$BFR             ; GET START OF CTB LIST
                                        27$:    MOV     $BFR,R0                 ; STORE NEXT CTB ADDRESS
                                                BEQ     5$                      ; IF END OF LIST - BRANCH
```

```
    801 002162                                  XKRB:
    802 002162  012701  000000G                         MOV     #$KRBLN,R1      ; GET LENGTH OF KRB ALLOCATION
    803
    804                                                 .IF DF  R$$MPL                                                  ;[MP01]
    805                                                 ADD     #K.PRM,R0       ; GET FIRST ADDRESS ALLOCATED TO KRB    ;[MP01]
    806                                                 .ENDC                                                           ;[MP01]
    807
    808 002166                                          CALL    $DEA16          ; DE-ALLOCATE THE KRB ALLOCATION
    809 002172                                          RETURN
    810
```

```
004054  001363                          BNE     10$             ;       ...


004056  016705  000000G         30$:    MOV     $PDVTA,R5       ; GET PDV TABLE ADDRESS
004062  016716  000000G                 MOV     $PDVNM,(SP)     ; AND COUNT OF PDV'S
004066  001415                          BEQ     52$             ; IF ZERO, NO WORK HERE
004070  012504                  40$:    MOV     (R5)+,R4        ; GET PDV ADDRESS
004072  001411                          BEQ     50$             ; ALLOW EMPTY SLOTS
004074  005764  000000G                 TST     Z.PCB(R4)       ; IS PROCESS CORE-RESIDENT?
004100  001406                          BEQ     50$             ; IF EQ, NO
004102  010467  000000G                 MOV     R4,$PDVA        ; MAKE SURE THAT $PDVA HAS THE PDV ADDRESS
004106                                  CALL    XDCB            ; POSSIBLE UNLOAD OF RSX11M DEVICE DRIVER
004112                                  CALL    XPRO            ; RELEASE PROCESS FROM MEMORY
004116  005316                  50$:    DEC     (SP)            ; LOOP <$PDVNM> TIMES
004120  001363                          BNE     40$             ;       ...
004122  012767  000002  000000G 52$:    MOV     #2,$RSIZE       ; SET TRANSFER SIZE
004130  016701  000000G                 MOV     $CCBSZ,R1       ; GET CCB SIZE

                                        .IF     DF,R$$MPL
                                        TST     .MPRO           ; MULTIPROCESSOR SYSTEM?
                                        BLE     53$             ; BR IF NO
                                        ADD     #2,R1           ; CCB IS 2 BYTES LONGER
                                        .ENDC

004134  016700  000000G         53$:    MOV     $CCBLH,R0       ; GET FREE CCB ADDRESS
004140  001412                          BEQ     56$             ; IF ZERO, NO MORE CCB'S
004142                                  GETRV   R0              ; READ IN CCB LINK WORD
004152  016767  000000G 000000G         MOV     $BFR,$CCBLH     ; SET NEW FIRST IN LIST

                                        .IF     DF,R$$MPL
                                        TST     .MPRO           ; MULTIPROCESSOR SYSTEM?
                                        BLE     54$             ; BR IF NO
                                        SUB     #2,R0           ; CCB HAS TWO EXTRA BYTES IN FRONT
                                54$:    .ENDC

004160                                  CALL    $DEA16          ; DE-ALLOCATE A CCB
004164  000763                          BR      53$             ; KEEP LOOPING
004166  005067  000000G         56$:    CLR     $RDBLH          ; CLEAR OUT LISTHEADS
004172  005067  000002G                 CLR     $RDBLH+2        ;    FOR BUFFER ALLOCATIONS
004176  005067  000000G                 CLR     $SDBLH          ;       ...
004202  005067  000002G                 CLR     $SDBLH+2        ;       ...
004206  005726                          TST     (SP)+           ; PURGE STACK

004210                                  CALL    XOBJTB          ; DEALLOCATE OBJECT TABLES

004214  005767  000000G                 TST     $DECPT          ; PSI ONLY SYSTEM ?
004220  001404                          BEQ     562$            ; IF YES - BRANCH

004222                                  CALL    XALITB          ; DEALLOCATE ALIAS TABLES
004226                                  CALL    XREMTB          ; DEALLOCATE REMOTE TABLES

004232                          562$:   CALL    XFLTTB          ; DEALLOCATE FILTER BLOCK TABLES

004236  005727  000000G                 TST     #$XPSI          ; IS PSI SUPPORT INCLUDED?
004242  001402                          BEQ     565$            ; BR IF NO
004244                                  CALL    $XPSI           ; ELSE UNLOAD PSI
004250  005727  000000G         565$:   TST     #$XSNA          ; Is SNA support included?
```

```
                                                                      L  1
VROB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 4
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE             REFERENCES

LX.NOD  = 000015            #5-63
LX.OBJ  = 000023            #5-63
LX.PRO  = 000005            #5-63
L.CTL   = ******   GX        29-1095    34-1270
L.DDM   = ******   GX        24-891     29-1093    34-1258
L.DDS   = ******   GX        22-754
L.DLC   = ******   GX        19-625     24-888     24-891
L.FLG   = ******   GX        22-745     23-774     24-817     24-846     24-896     25-921
L.NSTA  = ******   GX        14-424     14-430     19-623
L.UNT   = ******   GX        29-1097    29-1103    34-1280
MS.ALL  = ******   GX        20-668
MS.CSR  = ******   GX        20-678
MS.EXT  = ******   GX        16-502     16-514
MS.INC  = ******   GX        16-502     16-508
MS.LLC  = ******   GX        17-538
MS.PRI  = ******   GX        20-684     22-756
MS.SEC  = ******   GX        20-701
MS.VCT  = ******   GX        20-693
NOP       000120 R           6-133     #10-301
NTINIT    002646 R           26-953    #26-980
NULL      000234 R          #7-168      16-505     20-671
OF$INS  = 000001            #5-63
OF$LOG  = 000100            #5-63
OF$OFF  = 000002            #5-63
OF$ON   = 000000            #5-63
OF$SMC  = 000200            #5-63
OP$ADD  = 000200            #5-63
OP$ALL  = 000200            #5-63       22-757
OP$CCS  = 000004            #5-63
OP$CIE  = 004000            #5-63
OP$CON  = 000400            #5-63
OP$COP  = 000001            #5-63
OP$COS  = 000100            #5-63
OP$DST  = 000200            #5-63
OP$DTE  = 000040            #5-63
OP$DUP  = 000002            #5-63
OP$EST  = 000100            #5-63       14-396
OP$EVE  = 010000            #5-63
OP$FIL  = 000200            #5-63
OP$GRP  = 000100            #5-63
OP$HOS  = 000001            #5-63
OP$HTM  = 000002            #5-63
OP$INA  = 004000            #5-63
OP$INC  = 001000            #5-63
OP$KDS  = 002000            #5-63
OP$KDT  = 000400            #5-63
OP$KEV  = 001000            #5-63
OP$KGR  = 001000            #5-63
OP$KNT  = 010000            #5-63
OP$KSK  = 002000            #5-63
OP$LCT  = 000200            #5-63
OP$LNF  = 000040            #5-63       18-565
```

```
                                                                      M  1
VROB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 5
```

```
                                                                 L  2
VSET        CREATED BY  MACRO  ON 29-JUN-85 AT 02:59      PAGE 1

SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE              REFERENCES

F.NRBD  = ******  GX       *5-102      *5-112
F.RSIZ  = ******  GX       *5-103
F.URBD  = ******  GX       *5-113
SYSFDB  = ******  GX       *5-102      *5-103      *5-112      *5-113
$BFR    = ******  GX        5-112       5-113
$CETDF  = ******  GX        5-98
$CSRLD    000000 RG        #5-74
$NTLHB  = ******  GX        5-76
$PUTAD  = ******  GX        5-103
$SLTA   = ******  GX        5-89        5-97
$UNBLK  = ******  GX        5-80        5-102
$VUNB   = ******  GX        5-83
$$CSR   = ******  GX        5-99
.CXCSR  = ******  GX        5-76
```

```
                                                                 M  2
VSET        CREATED BY  MACRO  ON 29-JUN-85 AT 02:59      PAGE 2
```

```
   406
   407 000000                              STATE$  SELI,&P$SELI
   408 000000                               TRAN$  &K$ALL,          DONE,         ,             OP$ALL,       $OPTON
   409 000000                               TRAN$  $LAMDA,          SELIOP
   410                                     ;
   411                                     ;      SET LINE OPTIONS
   412                                     ;
   413 000000                              STATE$  SELIOP
   414 000000                               TRAN$  !OPCCSR,         SELIOP,       ,             OP$CCS,       $OPTON
   415 000000                               TRAN$  !OPDUP,          SELIOP,       ,             OP$DUP,       $OPTON
   416 000000                               TRAN$  !OPLOC,          SELIOP,       ,             OP$LOC,       $OPTON
   417 000000                               TRAN$  !OPMDEA,         SELIOP,       ,             OP$MDE,       $OPTON
   418 000000                               TRAN$  !OPPRI,          SELIOP,       ,             OP$PRI,       $OPTON
   419 000000                               TRAN$  !OPUCSR,         SELIOP,       ,             OP$UCS,       $OPTON
   420 000000                               TRAN$  !OPVEC,          SELIOP,       ,             OP$VEC,       $OPTON
   421 000000                               TRAN$  $LAMDA,          DONE
   422
   423                        ;
   424                        ; SET LOGGING
   425                        ;
   426 000000                              STATE$  SELO,&P$SELO
   427 000000                               TRAN$  $LAMDA,          SELOOP
   428                        ;
   429                        ;     SET LOGGING OPTIONS
   430                        ;
   431 000000                              STATE$  SELOOP
   432 000000                               TRAN$  !OPLGST,         SELOOP,       ,             OP$LST,       $OPTON
   433 000000                               TRAN$  !OPSNAM,         SELOOP,       ,             OP$NAM,       $OPTON
   434 000000                               TRAN$  !OPEVE,          SELOOP,       ,             OP$EVE,       $OPTON
   435 000000                               TRAN$  !OPSIN,          SELOOP,       ,             OP$SIN,       $OPTON
   436 000000                               TRAN$  !OPLIN,          SELOOP,       ,             OP$LNE,       $OPTON
   437 000000                               TRAN$  !OPNOD,          SELOOP,       ,             OP$NOD,       $OPTON
   438 000000                               TRAN$  !OPCIR,          SELOOP,       ,             OP$CIE,       $OPTON
   439 000000                               TRAN$  $LAMDA,          DONE,         $CKLOG
   440                        ;
   441                        ; SET NODE
   442                        ;
   443 000000                              STATE$  SENO,&P$SENO
   444 000000                               TRAN$  $LAMDA,          SENOOP
   445                        ;
   446                        ;     SET NODE OPTIONS
   447                        ;
   448 000000                              STATE$  SENOOP
   449 000000                               TRAN$  !OPADD,          SENOOP,       $IFNNA,       OP$ADD,       $OPTON
   450 000000                               TRAN$  !OPCIR,          SENOOP,       $IFNNA,       OP$NLI,       $OPTON
   451 000000                               TRAN$  !OPHOS,          SENOOP,       ,             OP$HOS,       $OPTON
   452 000000                               TRAN$  !OPNAM,          SENOOP,       ,             OP$NAM,       $OPTON
   453 000000                               TRAN$  !OPRPAS,         SENOOP,       ,             OP$RPA,       $OPTON
   454 000000                               TRAN$  !OPNOST,         SENOOP,       ,             OP$EST,       $OPTON
   455 000000                               TRAN$  !OPTPAS,         SENOOP,       ,             OP$TPA,       $OPTON
   456 000000                               TRAN$  !OPVER,          SENOOP,       ,             OP$VER,       $OPTON
   457 000000                               TRAN$  !OPSEG,          SENOOP,       ,             OP$SEG,       $OPTON
   458 000000                               TRAN$  !OPINC,          SENOOP,       ,             OP$INC,       $OPTON
   459 000000                               TRAN$  !OPOUT,          SENOOP,       ,             OP$OUT,       $OPTON
   460 000000                               TRAN$  !OPINA,          SENOOP,       ,             OP$INA,       $OPTON
   461 000000                               TRAN$  !OPRET,          SENOOP,       ,             OP$RET,       $OPTON
   462 000000                               TRAN$  $LAMDA,          DONE
```

```
  1070 000000                        STATE$  OPGR01
  1071 000000                         TRAN$   &K$GRPS,       $EXIT,      ,          OP$KGR,       $OPTON
  1072                       ;
  1073                       ; HELLO TIMER
  1074                       ;
  1075 000000                        STATE$  OPHTM
  1076 000000                         TRAN$   !HELTIM
  1077 000000                        STATE$  ,&P$HTM
  1078 000000                         TRAN$   $DNUMB,        $EXIT,      $HTM
  1079                       ;
  1080                       ; HOST
  1081                       ;
  1082 000000                        STATE$  OPHOS
  1083 000000                         TRAN$  &K$HOS
  1084 000000                        STATE$  ,&P$HOS
  1085 000000                         TRAN$  $DNUMB,         ,           $ARAD1
  1086 000000                        STATE$
  1087 000000                         TRAN$  <'.>,           OPHA
  1088 000000                         TRAN$  $LAMDA,         $EXIT,      $STHOS
  1089 000000                        STATE$  OPHA
  1090 000000                         TRAN$  $DNUMB,         ,           $ARAD2
  1091 000000                        STATE$
  1092 000000                         TRAN$  $LAMDA,         $EXIT,      $STHOS
  1093
  1094
  1095                       ;
  1096                       ; LINE
  1097                       ;
  1098 000000                        STATE$  OPLIN
  1099 000000                         TRAN$  &K$LIN
  1100 000000                        STATE$  ,&P$LNID
  1101 000000                         TRAN$  !LINID,         $EXIT
  1102                       ;
  1103                       ; LINE STATE
  1104                       ;
  1105 000000                        STATE$  OPLNST
  1106 000000                         TRAN$  &K$STAE
  1107 000000                        STATE$  ,&P$LIST
  1108 000000                         TRAN$  &K$OFF,         $EXIT
  1109 000000                         TRAN$  &K$ON,          $EXIT,      ,          VF.ON,        $VFLAG
  1110                       ;
  1111                       ; LISTEN TIMER
  1112                       ;
  1113 000000                        STATE$  OPLTM
  1114 000000                         TRAN$   !LISTIM
  1115 000000                        STATE$  ,&P$LTM
  1116 000000                         TRAN$   $DNUMB,        $EXIT,      $LTM
  1117                       ;
  1118                       ; LOCATION
  1119                       ;
  1120 000000                        STATE$  OPLOC
  1121 000000                         TRAN$  &K$LOC,         OPLOC1
  1122 000000                         TRAN$  &K$FIR,         $EXIT
  1123 000000                         TRAN$  &K$TOP,         $EXIT,      ,          LS.TOP,       RQBST
  1124 000000                        STATE$  OPLOC1,&P$LOC
  1125 000000                         TRAN$  &K$FIR,         $EXIT
  1126 000000                         TRAN$  &K$TOP,         $EXIT,      ,          LS.TOP,       RQBST
```

```
                                         .SBTTL  PEMSTA - ACTION ROUTINE FOR THE PROTOCOL EMULATOR STATE
                                 ;+
                                 ; **-PEMSTA-PROTOCOL EMULATOR STATE ACTION ROUTINE
                                 ;
                                 ; INPUTS:
                                 ;       NONE.
                                 ;
                                 ; OUTPUTS:
                                 ;       NONE.
                                 ;
                                 ;-

000040                           PEMSTA:
000040  052767  000014  000000G          BIS     #LS.NOD,RQBST           ; FAKE THIS AS A NODE FUNCTION
000046  052767  000000G 000000G          BIS     #VF.PEM,$VFLAG          ; ..BUT INDICATE IT AS A PROTOCOL EMUL.
000054                                   CALL    $EXNOD                  ; FAKE EXECUTOR NODE
000060                                   RETURN
```

```
SYMBOL CROSS REFERENCE                                         CREF    04.00

SYMBOL  VALUE            REFERENCES

$LTM    = ******  G        5-93
$MACT   = ******  G        5-93
$MDEA   = ******  G        5-93
$MXCO   = ******  G        5-94
$MXLI   = ******  G        5-94
$NADD   = ******  G        5-94
$NNAM   = ******  G        5-94
$NUMBR  = 000002          #6-108
$ONAM   = ******  G        5-94
$OPTON  = ******  G        5-84
$OTYP   = ******  G        5-94
$OUT    = ******  G        5-100
$OVER   = ******  G        5-95
$OWN    = ******  G        5-95
$PCIR   = ******  G        5-95
$PDT    = ******  G        5-95
$PNAM   = ******  G        5-95
$PRGCL  = ******  G        5-95
$PRI    = ******  G        5-95
$QUAL   = ******  G        5-84
$RAD50  = 000016          #6-108
$RET    = ******  G        5-100
$RONLY  = 000001          #5-67        6-108     6-108     6-108     6-108
$SCOND  = ******  G        5-96
$SCOP1  = ******  G        5-96
$SCOP2  = ******  G        5-96
$SCOP3  = ******  G        5-96
$SCSR   = ******  G        5-96
$SEG    = ******  G        5-100
$STADD  = ******  G        5-96
$STALI  = ******  G        5-97
$STDES  = ******  G        5-97
$STEND  = ******  G        5-97
$STHOS  = ******  G        5-97
$STLND  = ******  G        5-97
$STLSH  = ******  G        5-96
$STLSN  = ******  G        5-97
$STNAD  = ******  G        5-98
$STNAM  = ******  G        5-98
$STPAR  = ******  G        5-98
$STRNG  = 000004          #6-108
$STRPA  = ******  G        5-98
$STSNM  = ******  G        5-98
$STTPA  = ******  G        5-98
$SUBXP  = 000010          #6-108
$TRI    = ******  G        5-99
$TRIBA  = ******  G        5-99
$TYPE   = ******  G        5-84
$UNT    = ******  G        5-99
$VEC    = ******  G        5-99
$VFLAG  = ******  G        5-84        *19-1642
$XMISS  = ******  G        5-99
```

```
Symbol table

AB     = 000000 G      LF.X2P= 000000 G       N$$MOV= 000010         ZF.DDM= 000001 G       .CXVEC  000026 G
A$$CHK= 000000         LN.CLO= 000000 G       N$$NCT= 000001         ZF.DIA= 004000 G       .CXVSM  000016 G
A$$CPS= 000000         LN.DUM= 000005 G       N$$PEM= 000001         ZF.DLC= 000002 G       .DPRB   000041 G
A$$PRI= 000000         LN.LOA= 000004 G       PIRQ   = 177772 G      ZF.DVP= 100000 G       .DPRW   000042 G
A$$TRP= 000000         LN.LOO= 000003 G       PMODE  = 030000 G      ZF.INI= 040000 G       .DVCHA  000011 G
BUFUMP= 172354 G       LN.OAU= 000003 G       PR0    = 000000 G      ZF.KMX= 000020 G       .EOF    000012 G
CMODE  = 140000 G      LN.OFF= 000001 G       PR1    = 000040 G      ZF.LLC= 000004 G       .FILE   000013 G
C$$CKP= 000000         LN.ON  = 000000 G      PR2    = 000100 G      ZF.LMC= 000100 G       .INT    000014 G
C$$ORE= 000400         LN.OOP= 000004 G       PR3    = 000140 G      ZF.MAN= 020000 G       .LFILE  000017 G
C$$RSH= 177564         LN.OPE= 000001 G       PR4    = 000200 G      ZF.MFL= 000010 G       .LIBR   000015 G
D$$BUG= 177514         LN.REF= 000002 G       PR5    = 000240 G      ZF.MTM= 000400 G       .LINKS  000016 G
D$$ISK= 000000         LN.SER= 000002 G       PR6    = 000300 G      ZF.MUX= 000040 G       .LSTHD  000020 G
D$$L11= 000001         LN.STA= 000017 G       PR7    = 000340 G      ZF.PSE= 002000 G       .LTAB   000021 G
D$$YNC= 000000         LN.SUB= 000360 G       PS     = 177776 G      ZF.SLI= 010000 G       .MPTAB  000022 G
D$$YNM= 000000         LN.TRI= 000006 G       P$$P45= 000000         ZF.TIM= 000200 G       .MXPTB  000044 G
E$$XPR= 000000         L$$ASG= 000000         P$$WRD= 000000         ZF.X3P= 000000 G       .MXTAB  000023 G
F$$LVL= 000001         L$$DRV= 000000         Q$$OPT= 000010         ZS.ASN= 100000 G       .NODE   000024 G
G$$TPP= 000000         L$$P11= 000001         R$$DER= 000000         ZS.BSY= 140000 G       .PECHA  000045 G
G$$TSS= 000000         L$$11R= 000000         R$$K11= 000001         Z.AVL   000014 G       .PRI    000025 G
G$$TTK= 000000         L.COST  000015 G       R$$SND= 000000         Z.DAT   000016 G       .PRTOV  000026 G
G$$WRD= 000000         L.CTL   000012 G       R$$11M= 000000         Z.DSP   000000 G       .SBPCB  000066 G
I$$RAR= 000000         L.CVA   177776 G       SF.ACT= 000200 G       Z.FLG   000010 G       .SCOM   000027 G
I$$RDN= 000000         L.DDM   000002 G       SF.ENA= 000100 G       Z.LEN = 000016 G       .SECSR  000030 G
KISAR0= 172340 G       L.DDS   000004 G       SF.LPB= 000004 G       Z.LLN   000006 G       .SLNB   000031 G
KISAR6= 172354 G       L.DLC   000003 G       SF.MFL= 000040 G       Z.MAP   000020 G       .SLNW   000032 G
K$$CNT= 177546         L.DLM   000006 G       SF.PAC= 000020 G       Z.NAM   000004 G       .STNB   000033 G
K$$CSR= 177546         L.DLS   000010 G       SF.REA= 000010 G       Z.PCB   000012 G       .STNW   000034 G
K$$LDC= 000000         L.FLG   000000 G       SF.SER= 000001 G       Z.SCH   000007 G       .TIME   000043 G
K$$TPS= 000074         L.KRBA  000016 G       SF.SVC= 000002 G       .ADDRB  000001 G       .UNB    000035 G
LB.ADR  000006 G       L.LEN = 000022 G       SF.UNL= 000040 G       .ADDRW  000002 G       .UNW    000036 G
LB.APR= 000013 G       L.MPF   000022 G       SWR    = 177570 G      .APR    000040 G       .WORD   000037 G
LB.SIZ  000010 G       L.NMST  000020 G       S$$WRG= 000000         .BASEB= 140000 G       .X2CH1  000061 G
LB.USE  000012 G       L.NSTA  000014 G       S$$YSZ= 007600         .BLKB   000003 G       .X2CH2  000062 G
LD$LP  = 000000        L.OWNR  000021 G       S.COST  000001 G       .BLKW   000004 G       .X2CH3  000063 G
LF.ACT= 100000 G       L.UNT   000013 G       S.FLG   000000 G       .BYTE   000005 G       .X2CH4  000064 G
LF.BRO= 000400 G       MPAR   = 172100 G      S.LEN   000004 G       .CORE   000006 G       .X2CH5  000065 G
LF.BWT= 000007 G       MPCSR  = 177746 G      S.NMST  000002 G       .CSR    000007 G       .X3C01  000046 G
LF.ENA= 002000 G       M$$CRB= 000124         S.OWNR  000003 G       .CTIM   000010 G       .X3C02  000047 G
LF.LPB= 001000 G       M$$CRX= 000000         TPS    = 177564 G      .CXALL  000062 G       .X3C03  000050 G
LF.MDC= 000100 G       M$$FCS= 000000         T$$KMG= 000000         .CXCSR  000004 G       .X3C04  000051 G
LF.MFL= 004000 G       M$$MGE= 000000         T$$MIN= 000000         .CXLBR  000060 G       .X3C05  000052 G
LF.MTP= 000020 G       M$$NET= 000000         UBMPR  = 170200 G      .CXLB1  000030 G       .X3C06  000053 G
LF.PAC= 000200 G       M$$OVR= 000000         UISAR0= 177640 G       .CXLB2  000044 G       .X3C07  000054 G
LF.RDY= 040000 G       N$$ACC= 000001         UISAR1= 177642 G       .CXLEN  000064 G       .X3C08  000055 G
LF.REA= 010000 G       N$$BUF= 000001         V$$CTR= 001000         .CXPCB  000006 G       .X3C09  000056 G
LF.SER= 000040 G       N$$LDV= 000001         X$$DBT= 000000         .CXSYM  000010 G       .X3C10  000057 G
LF.TIM= 000010 G       N$$MCP= 000001         ZF.COU= 001000 G       .CXUNL  000024 G       .X3C11  000060 G
LF.UNL= 020000 G       N$$MLL= 000001

. ABS.  177776     000   (RW,I,GBL,ABS,OVR)
        000000     001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
```

```
                                                .SBTTL  ERROR MESSAGES

                                                .NLIST  BEX

                                        ;
                                        ; ERROR MESSAGE BUFFER
                                        ;
000560      015                         ERRMSG: .BYTE   15
000561      126     116     120                 .ASCII  "VNP "
000565                                  ERRBUF: .BLKB   73.

                                        ;
                                        ; ERROR MESSAGE PREFIXES
                                        ;
000676      055     055     040         WPFX:   .ASCII  "-- *WARNING* "
000713      055     055     040         EPFX:   .ASCIZ  "-- "

                                                .EVEN

                                        ;
                                        ; ERROR MESSAGES
                                        ;
000720                                          NTLER$  ,13,2,$TERR,$REP.C,$ELINE,<End of task buffer overflow.>
000766                                          NTLER$  ,21,2,$TERR,$REP.C,$ELINE,<DDM operator>
001014                                          NTLER$  ,79,2,$TERR,$REP.C,$ELINE,<DLC operator>
001042                                          NTLER$  ,82,2,$TERR,$REP.C,$ELINE,<Operator may not occur twice.>
001110                                          NTLER$  ,83,2,$TERR,$REP.C,$ELINE,<Primary template operator.>
                                        ;
                                        ; ERROR MESSAGE FORMAT STRINGS
                                        ;
001154      052     040     124         FMT2:   .ASCIZ  "* Template -- "
                                                .EVEN

000000                                          .PSECT
```

```
    593                                         .SBTTL  $TERR - PRINT ERROR MESSAGE
    594                                         .SBTTL  $TWARN - PRINT WARNING
    595                                 ;+
    596                                 ; $TERR - PRINT ERROR MESSAGE
    597                                 ; $TWARN - PRINT WARNING MESSAGE
    598                                 ;
    599                                 ; INPUTS:
    600                                 ;       R0=ERROR MESSAGE BLOCK ADDRESS
    601                                 ;
    602                                 ; OUTPUTS:
    603                                 ;       C-BIT=SET
    604                                 ;       R0,R1=DESTROYED
    605                                 ;-
    606
    607                                         .ENABL  LSB
    608
    609 001444                          $TWARN::
    610 001444  012701  000676'                 MOV     #WPFX,R1        ; SET PREFIX ADDRESS
    611 001450  005767  000000G                 TST     $FLAGS          ; ARE WARNING MESSAGES ENABLED?
    612 001454  100412                          BMI     10$             ; IF MI, YES
    613 001456  000445                          BR      60$             ; ELSE JUST EXIT WITH C-BIT SET
    614 001460                          $TWE::
    615 001460  032767  000000G 000000G         BIT     #FL.LLC,$FLAGS  ; WARNING IF LLC
    616 001466  001366                          BNE     $TWARN          ; FATAL IF DDM OR DLC
    617 001470                          $TERR::
    618 001470  052767  100000  000000'         BIS     #F1.ERR,$FLAG1  ; INDICATE ERROR HAS OCCURRED
    619 001476  012701  000713'                 MOV     #EPFX,R1        ; SET PREFIX ADDRESS
    620 001502                          10$:    CALL    $SAVAL          ; SAVE ALL REGISTERS
    621 001506  010005                          MOV     R0,R5           ; COPY ERROR MESSAGE BLOCK ADDRESS
    622 001510  012704  000565'                 MOV     #ERRBUF,R4      ; PLACE TO BUILD ERROR MESSAGE
    623 001514  112124                  20$:    MOVB    (R1)+,(R4)+     ; COPY THE PREFIX
    624 001516  001376                          BNE     20$             ;       ...
    625 001520  005304                          DEC     R4              ; SKIP THE ZERO BYTE
    626 001522  012503                          MOV     (R5)+,R3        ; GET FORMAT STRING ADDRESS
    627 001524                                  CALL    @(R5)+          ; DO THE FORMATTING
    628 001526  105713                          TSTB    (R3)            ; IS IT END OF FORMAT STRING?
    629 001530  001403                          BEQ     40$             ; IF EQ, YES
    630 001532  112324                  30$:    MOVB    (R3)+,(R4)+     ; ELSE COPY REST OF FORMAT STRING
    631 001534  001376                          BNE     30$             ;       ...
    632 001536  005304                          DEC     R4              ; SKIP THE ZERO BYTE
    633 001540  012503                  40$:    MOV     (R5)+,R3        ; GET LINE 2 ADDRESS
    634 001542  112524                  50$:    MOVB    (R5)+,(R4)+     ; COPY THE ERROR TEXT
    635 001544  001376                          BNE     50$             ;       ...
    636 001546  005304                          DEC     R4              ; SKIP THE ZERO BYTE
    637 001550  012700  000560'                 MOV     #ERRMSG,R0      ; GET ERROR MESSAGE ADDRESS
    638 001554  010401                          MOV     R4,R1           ; COPY END ADDRESS
    639 001556  160001                          SUB     R0,R1           ; GIVING LENGTH
    640 001560                                  CALL    $TIOUT          ; PRINT IT
    641 001564  005703                          TST     R3              ; SHOULD SECOND LINE BE PRINTED?
    642 001566  001401                          BEQ     60$             ; IF EQ, NO
    643 001570                                  CALL    (R3)            ; PRINT SECOND ERROR MESSAGE LINE
    644 001572  000261                  60$:    SEC                     ; RETURN C-BIT SET
    645 001574                                  RETURN
    646
    647                                         .DSABL  LSB
    648
    649                 000001                  .END
```

```
                                          .SBTTL  $TM.IN - INITIALIZE TEMPLATE DATA
                                  ;+
                                  ; $TM.IN - INITIALIZE TEMPLATE DATA, OPEN PRIMARY TEMPLATE
                                  ;
                                  ; INPUTS:
                                  ;       NONE
                                  ;
                                  ; OUTPUTS:
                                  ;       C-BIT=SUCCESS/FAILURE
                                  ;       $ADDR=LINE TABLE ADDRESS IF MUX UPDATE
                                  ;       R0,R1,R2=DESTROYED
                                  ;-
                                          .ENABL  LSB
000000                            $TM.IN::
000000   016767  000000G 000000G          MOV     $XBUF2,$BIN      ; COPY END OF TASK BUFFER ADDRESS
000006   016767  000000G 000000G          MOV     $XBUF2,$PTR      ; TWICE
000014   016767  000000G 000000G          MOV     $XLEN2,$LEFT     ; COPY BUFFER SIZE
000022   005067  000000G                  CLR     $ADDR            ; CLEAR LINE TABLE ADDRESS
000026   005067  000000G                  CLR     $MXADD           ; CLEAR MUX LINE TABLE ADDRESS
000032   005067  000000G                  CLR     $MXSIZ           ; CLEAR MUX TABLE SIZE
000036   005067  000000G                  CLR     $OFF             ; CLEAR LINE TABLE OFFSET
000042   005067  000000G                  CLR     $SIZE            ; CLEAR LINE TABLE SIZE
000046   012767  000000G 000000G          MOV     #F1.UMR,$FLAG1   ; INITIALIZE LINE TABLE FLAGS
000054   005067  000000G                  CLR     $NVECT           ; CLEAR NUMBER OF VECTORS
000060   005067  000000G                  CLR     $NALL            ; CLEAR NUMBER OF ALLOCATIONS
000064   012767  000000G 000000G          MOV     #$SZPTR,$SZPTR   ; INITIALIZE SIZE TABLE POINTER
000072   016700  000000G                  MOV     $NAME,R0         ; GET THREE CHARACTER PDV NAME
000076                                    CALL    $DFN             ; SETUP DEFAULT FILE NAME
000102   016701  000000G                  MOV     $PDVA,R1         ; GET PDV ADDRESS
000106   032761  000000G 000000G          BIT     #ZF.PSE,Z.FLG(R1) ; IS THIS A PSEUDO DDM?
000114   001051                           BNE     10$              ; BR IF YES
000116   032767  000000G 000000G          BIT     #FL.DDM,$FLAGS   ; IS THIS A DDM LINE TABLE ?
000124   001445                           BEQ     10$              ; NO ..
000126   016700  000000G                  MOV     $SLTA,R0         ; GET THE SLT ADDRESS
000132   016067  000000G 000000G          MOV     L.DDS(R0),$ADDR  ; HAS THE LINE TABLE ALREADY BEEN BUILT ?
000140   001437                           BEQ     10$              ; NO ..
000142   052767  000000G 000000G          BIS     #FL.CHA,$FLAGS   ; YES .. IT'S CHARACTERISTICS UPDATE
000150   016000  000000G                  MOV     L.KRBA(R0),R0    ; GET THE KRB ADDRESS

                                          .IF DF  R$$MPL
                                          ADD     #K.PRM,R0        ; POINT TO START OF KRB
                                          GETRV   R0,#$KRBLN       ; READ KRB
                                          .IFF
000154                                    GETRV   R0,#K.CSR+2      ; READ IN KRB

                                          .ENDC

                                          .IF DF  R$$MPL
                                          CMP     $$CSR,K.CSR-K.PRM+$BFR ; SAME CSR ADDRESS ?
                                          .IFF
000172   026767  000000G 000002G          CMP     $$CSR,K.CSR+$BFR ; IS THIS THE SAME CSR ADDRESS ?

                                          .ENDC

000200   001403                           BEQ     4$               ; IF EQ, YES .. OKAY
000202                                    EMSG$   WZ               ; WARNING - MUX UPDATE ONLY

```

```
    172                                                  .SBTTL  $VTST - VNP TESTING ROUTINES
    173                                         ;+
    174                                         ; INPUTS:
    175                                         ;        NONE.
    176                                         ;
    177                                         ; OUTPUTS:
    178                                         ;        COMMANDS PERFORMED AS REQUESTED.
    179                                         ;
    180                                         ;-
    181                                                  .ENABL  LSB
    182 000036                                  $VTST::
    183
    184 000036                                  TST:     QREAD   <VDT>
    185 000046  103407                                   BCS     EXIT             ; IF CS, EOF
    186 000050  032767  000000G 000000G                  BIT     #VF.VDT,$VFLAG   ; SHOULD BUFFERS BE OUTPUT?
    187 000056  001016                                   BNE     5$               ; NO IF NE
    188 000060                                           CALL    BUFOU            ; YES, SO OUTPUT BUFFERS
    189 000064  000413                                   BR      5$
    190 000066  042767  000000G 000000G EXIT:    BIC     #VF.VDT,$VFLAG   ; CLEAR BIT SO WILL OUTPUT NEXT TIME
    191 000074  032767  000000G 000000G          BIT     #FE.CEX,$FMASK   ; DO WE NEED TO READ THE IMAGE BACK IN ?
    192 000102  001403                                   BEQ     1$               ; IF EQ, NO
    193 000104  005002                                   CLR     R2               ; NO RELOCATION BIAS
    194 000106                                           CALL    $VIMI            ; ELSE, READ IN IMAGE
    195 000112                                  1$:      RETURN
    196
    197 000114  012700  000674'                 5$:      MOV     #VECTOR,R0
    198 000120  016705  000000G                 10$:     MOV     $CLBUF,R5
    199 000124  122025                                   CMPB    (R0)+,(R5)+
    200 000126  001005                                   BNE     19$
    201 000130  122025                                   CMPB    (R0)+,(R5)+
    202 000132  001004                                   BNE     20$
    203 000134                                           CALL    @(R0)
    204 000140  000736                                   BR      TST
    205 000142  105720                          19$:     TSTB    (R0)+
    206 000144  005720                          20$:     TST     (R0)+
    207 000146  001364                                   BNE     10$
    208                                                  .DSABL  LSB
    209
    210 000150                                           TYPES   <CHOOSE ONE OF THE FOLLOWING FUNCTIONS:>,DS
    211 000200                                           TYPES   <GV - GETRV (GET 16BIT ADDRESS)>,DS
    212 000230                                           TYPES   <GR - GETRC (GET 16BIT VIRTUAL ADDRESS)>
    213 000260                                           TYPES   <GA - GETAD (GET 22BIT ADDRESS)>
    214 000310                                           TYPES   <PR - PUTRC (PUT 16BIT VIRTUAL ADDRESS)>
    215 000340                                           TYPES   <PA - PUTAD (PUT 22BIT ADDRESS)>
    216 000370                                           TYPES   <A6 - ALL16 (ALLOCATE DSR)>
    217 000420                                           TYPES   <A2 - ALL22 (ALLOCATE NTPOOL STORAGE)>
    218 000450                                           TYPES   <D6 - DEA16 (DEALLOCATE 16BIT ADDRESS)>
    219 000500                                           TYPES   <D2 - DEA22 (DEALLOCATE 22BIT ADDRESS)>
    220 000530                                           TYPES   <DB - DUMP BYTES>
    221 000560                                           TYPES   <DW - DUMP WORDS>
    222 000610                                           TYPES   <MO - MODIFY>
    223 000640                                           TYPES   <OD - ODT>
    224 000670  000167  177142                           JMP     TST
    225
    226 000674     107     126                  VECTOR:  .ASCII  /GV/
    227 000676  001052'                                  .WORD   GETRV
    228 000700     107     122                           .ASCII  /GR/
```

```
     55                                         ;***
     56                                         ; LIBRARY MACROS
     57                                         ;***
     58                                                 .MCALL  GETRV,PUTRC,SCBDF$,SWTCHI,SWTCHO,PUTAD,GETAD
     59
     60                                         ;***
     61                                         ; LIBRARY SYMBOLS
     62                                         ;***
     63 000000                                          SCBDF$                  ; UMR BLOCK OFFSETS
     64
     65                                         ;***
     66                                         ; LOCAL DATA
     67                                         ;***
     68
     69 000000                                          .PSECT  DATA,D
     70
     71                                         ;
     72                                         ; LOCAL COPIES OF PREVIOUS AND NEXT UMR BLOCKS
     73                                         ;
     74 000000                                  PREV:   .BLKW   M.LGTH/2
     75 000014                                  NEXT:   .BLKW   M.LGTH/2
     76
     77                                         ;
     78                                         ; FIRST UNIBUS MAPPING REGISTER ADDRESS
     79                                         ;
     80          170200                                 UBMPR=170200
     81
     82 000000                                          .PSECT
```

```
105                                                       .MCALL  SWTCHI,SWTCHO,GETAD,PUTAD,ERMSG$,ERBLK$,ERRPT$
106
107                                                       .ENABL  LC,LSB
108
109 000000                                       $VUNB::
110 000000  004567  000000G                               JSR     R5,$SAVRG                   ; SAVE R3,R4,R5
111 000004  005077  000000G                               CLR     @$ADRZ                      ; ZERO THIS HALF
112 000010  010205                                        MOV     R2,R5                       ; SAVE ADDRESS FLAG
113 000012  001427                                        BEQ     52$                         ; IF ZERO, GO ALLOCATE NEW BLOCK
114 000014  011077  000000G                               MOV     (R0),@$ADRX                 ; DESIRED ADDRESS
115 000020  027767  000000G 000000G                       CMP     @$ADRX,$UNBAD               ; IS CURRENT BLOCK THE DESIRED ONE ?
116 000026  001024                                        BNE     53$                         ; NO
117
118 000030  020227  177702G                  50$:         CMP     R2,#$UNBLK-62.              ; IS ADDRESS LEGAL ?
119 000034  103545                                        BLO     102$                        ; NO
120 000036  060102                                        ADD     R1,R2                       ; INCREMENT TO NEXT SEGMENT ADDRESS
121 000040  010203                                        MOV     R2,R3                       ; COPY TENTATIVE SEGMENT ADDRESS
122 000042  060103                                        ADD     R1,R3                       ; DOES SEGMENT EXTEND PAST EOB ?
123 000044  022703  000000G                               CMP     #$UNEND,R3                  ; ...
124 000050  103037                                        BHIS    60$                         ; NO
125 000052  012700  000000G                               MOV     #$UNBAD,R0                  ; POINT AT DESIRED BLOCK ADDRESS
126 000056  016777  000000G 000000G                       MOV     $UNBLK,@$ADRX               ; LINK TO NEXT BLOCK
127 000064  001005                                        BNE     53$                         ; NO NEXT BLOCK
128 000066  005705                                        TST     R5                          ; IS A NEW BLOCK TO BE APPENDED ?
129 000070  001025                                        BNE     55$                         ; NO
130 000072                                   52$:         CALL    LINK                        ; CREATE NEW LINK
131 000076  000416                                        BR      54$
132 000100                                   53$:         SWTCHI  #$UNBLK                     ; SET UP BUFFER POINTERS
133 000106                                                SWTCHO  #$UNBLK                     ; ...
134 000114  017767  000000G 000000G                       MOV     @$ADRX,$UNBAD               ; STORE ADRRESS FOR REFERENCE
135 000122                                                GETAD   ,#64.                       ; READ IN BLOCK IMAGE
136 000134  012702  000002G                  54$:         MOV     #$UNBLK+2,R2                ; SET STARTING ADDRESS
137 000140  160102                                        SUB     R1,R2                       ; MINUS SEGMENT SIZE
138 000142  000732                                        BR      50$                         ; AND RESTART
139 000144  005002                           55$:         CLR     R2                          ; SIGNAL NO NEXT SEGMENT
140 000146  000261                                        SEC                                 ; ...
141
142 000150                                   60$:         SWTCHI                              ; REASSIGN
143 000156                                                SWTCHO                              ; DEFAULT BUFFERS
144
145 000164                                                RETURN
```

```
                                                GETRV   R0,#2           ; READ CTB
                                                CMP     R0,R1           ; IS THIS OUR CTB ?
                                                BEQ     26$             ; IF YES - BRANCH
                                                MOV     R0,R2           ; IF NO - COPY PREVIOUS ADDRESS
                                                BR      27$             ; CONTINUE

                                        26$:    CMP     $CTLST,R0       ; UPDATE LISTHEAD ?
                                                BNE     25$             ; IF NO BRANCH
                                                MOV     $BFR,$CTLST     ; UPDATE LISTHEAD
                                                BR      5$              ; CONTINUE

                                        25$:    SWTCHI
                                                SWTCHO
                                                CLC
                                                PUTRC   R2,#2           ; DELETE OUT CTB FROM LIST
                                                                                                        ; ...
                                                .ENDC                                                   ;[MP01]

000152  032767  000000G 000000G 5$:     BIT     #VF.SS,$VFLAG   ; 11S SYSTEM ?
000160  001477                          BEQ     XDLC            ; IF NO - BRANCH
000162  032764  000000G 000000G         BIT     #ZF.LMC,Z.FLG(R4) ; RELEASE MICROCODE PARTITION ?
000170  001473                          BEQ     XDLC            ; IF NO - BRANCH
000172  016767  000000G 000000G         MOV     $PARHD,$BFR     ; CHECK IF SERVER BASE
000200  005001                          CLR     R1              ; INIT PCB ADDRESS
000202  016700  000000G         1$:     MOV     $BFR,R0         ; GET NEXT PCB ADDRESS
000206  001460                          BEQ     3$              ; IF END OF LIST - NOT SERVER BASE
000210                                  GETRV   R0,.PLGTH       ; READ PCB
000226  022767  026226  000004G         CMP     #^RGEN,P.NAM+$BFR ; CHECK FOR MAIN PARTITION
000234  001034                          BNE     2$              ; ...
000236  022767  000000  000006G         CMP     #^R   ,P.NAM+2+$BFR ; ...
000244  001030                          BNE     2$              ; IF NO MATCH - BRANCH
000246  016700  000000G                 MOV     $BFR,R0         ; SAVE NEXT PCB ADDRESS

000252  016701  000010G         4$:     MOV     P.SUB+$BFR,R1   ; GET NEXT SUB-PARTITION ADDRESS
000256  001003                          BNE     6$              ; IF NOT END OF LIST - BRANCH
000260  010067  000000G                 MOV     R0,$BFR         ; RESTORE NEXT MAIN PCB ADDRESS
000264  000746                          BR      1$              ; CONTINUE

000266                          6$:     GETRV   R1,.PLGTH       ; READ PCB
000304  026767  000000G 000004G         CMP     $NAME,P.NAM+$BFR ; MATCH ?
000312  001357                          BNE     4$              ; IF NO - BRANCH
000314  022767  050670  000006G         CMP     #^RMC,P.NAM+2+$BFR ; MATCH ?
000322  001353                          BNE     4$              ; IF NO - BRANCH
000324  000411                          BR      3$

000326  022767  055260  000004G 2$:     CMP     #^RNTP,P.NAM+$BFR ; MATCH ?
000334  001322                          BNE     1$              ; IF NO - BRANCH
000336  022767  060044  000006G         CMP     #^ROOL,P.NAM+2+$BFR ; MATCH ?
000344  001316                          BNE     1$              ; IF NO - BRANCH
000346  000404                          BR      XDLC            ; IF YES - SERVER BASE SYSTEM

000350  010100                  3$:     MOV     R1,R0           ; DEALLOCATE MICROCODE PARTITION
000352  001402                          BEQ     XDLC            ; THIS BRANCH SHOULD NEVER OCCUR
000354                                  CALL    $XPAR           ; ...

000360  016400  000000G         XDLC::  MOV     Z.PCB(R4),R0    ; GET PCB ADDRESS
000364  001454                          BEQ     101$            ; IF NO PCB, LOGIC ERROR
```

```
                                                .SBTTL  FNDADD - FIND LINE TABLE ADDRESS
                                        ;+
                                        ; FNDADD - FIND LINE TABLE ADDRESS
                                        ;
                                        ; INPUTS:
                                        ;       R3 = VECTOR ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       R3 = LINE TABLE ADDRESS
                                        ;       R0 = DESTROYED
                                        ;
                                        ;-

002174  016700  000000G                 FNDADD: MOV     $ICBLN,R0               ; GET LENGTH OF AN ICB
                                                .IF NDF R$$MPL                                                  ;[MP01]
002200                                          GETRV   R3,#4                   ; GET VECTOR LOCATION
                                                .IFF                                                            ;[MP01]
                                                GETAD   R3,#4                   ; GET VECTOR LOCATION           ;[MP01]

                                                .ENDC

002216  016703  000000G                         MOV     $BFR,R3                 ; ASSUME THIS IS THE LINE TABLE

                                                .IF NDF R$$MPL                                                  ;[MP01]
002222                                          GETRV   R3,R0                   ; READ IN ICB OR LINE TABLE
                                                .IFF                                                            ;[MP01]
                                                BIT     #DF.DAS,$DFLAG          ; D-SPACE SYSTEM ?              ;[MP01]
                                                BNE     5$                      ; IF YES - BRANCH               ;[MP01]
                                                MOV     #ICBI,R0                ; SET UP I-SPACE ICB SIZE       ;[MP01]
                                        5$:     GETAD   R3,R0                   ; READ IN ICB                   ;[MP01]

                                                .ENDC                                                           ;[MP01]

002236  022760  004537  177774G                 CMP     #4537,$BFR-4(R0) ; IF "JSR R5,X", THEN THIS IS LINE TABLE
002244  001406                                  BEQ     10$                     ; ..ADDRESS
002246  022760  000137  177774G                 CMP     #137,$BFR-4(R0)  ; IF "JMP LINE-TABLE" THEN LINE TABLE ADDRESS
002254  001003                                  BNE     101$                    ; ..IS NEXT WORD, ELSE ERROR
002256  016003  177776G                         MOV     $BFR-2(R0),R3           ; RETURN REAL LINE TABLE ADDRESS
002262                                  10$:    RETURN

002264                                  101$:   CRASH                           ; WHAT IS R2 POINTING AT ??

                                        ;
                                        ; SCAN SLT FOR ACTIVITY
                                        ;
002266  005000                          SCAN:   CLR     R0                      ; OTHER READY LINES USING THIS DLC
002270  005001                                  CLR     R1                      ; OTHER READY LINES USING THIS DDM
002272  005067  000004'                         CLR     UNITS                   ; OTHER READY LINES ON THIS CONTROLLER
002276  016702  000000G                         MOV     $SLTMA,R2               ; GET SLT MAPPING TABLE ADDRESS
002302  016704  000000G                         MOV     $SLTNM,R4               ; GET NUMBER OF SLT'S
002306  012203                          10$:    MOV     (R2)+,R3                ; GET NEXT SLT
002310  032763  000000G 000000G                 BIT     #LF.RDY,L.FLG(R3) ; IS THIS LINE READY?
002316  001420                                  BEQ     30$                     ; IF EQ, NO
002320  126365  000000G 000000G                 CMPB    L.DLC(R3),L.DLC(R5) ; SAME DLC AS OURS?
002326  001001                                  BNE     20$                     ; IF NE, NO
002330  005200                                  INC     R0                      ; TALLY IT
002332  126365  000000G 000000G 20$:            CMPB    L.DDM(R3),L.DDM(R5) ; SAME DDM AS OURS?
```

```
   1354 004254  001402                                 BEQ     57$              ; Br if no
   1355 004256                                         CALL    $XSNA            ; Else unload SNA
   1356
   1357 004262  005067  000000G               57$:     CLR     $CXOPT           ; NO COMM EXEC OPTIONS SET NOW
   1358
   1359 004266  016700  000000C                        MOV     $NTLHB+.CXALL,R0 ; GET CETAB ALLOCATION
   1360 004272  016701  000000C                        MOV     $NTLHB+.CXLEN,R1 ; AND LENGTH
   1361 004276                                         CALL    $DEA16           ; DEALLOCATE THE CETAB ALLOCATION
   1362
   1363 004302  016702  000002G                        MOV     $NTLHB+2,R2      ; NTPOOL UMR BLOCK
   1364 004306  016703  000004G                        MOV     $NTLHB+4,R3      ; $DEUMR ADDRESS IN R3
   1365
   1366 004312  016700  000000G               58$:     MOV     $NTLHB,R0        ; NTPOOL PCB ADDRESS
   1367 004316                                         CALL    $XPAR            ; RELEASE NTPOOL PARTITION
   1368 004322  103002                                 BCC     465$             ; IF CC - OK
   1369 004324                                         CALLR   101$             ; GO CRASH
   1370
   1371 004330  016700  000000C               465$:    MOV     $NTLHB+.SBPCB,R0 ; GET SERVER BASE POOL PCB
   1372 004334  001405                                 BEQ     59$              ; IF NONE BRANCH
   1373 004336                                         CALL    $XPAR            ; DEALLOCATE PCB
   1374 004342  103002                                 BCC     59$              ; IF FAILURE - BRANCH
   1375 004344                                         CALLR   101$             ; GO CRASH
   1376 004350                                59$:
   1377
   1378                                                .IF     NDF,R$$MPL
   1379 004350  016700  000000C                        MOV     .CXPCB+$NTLHB,R0 ; GET CEX PCB ADDRESS
   1380 004354                                         CALL    $XPAR            ; RELEASE CEX FROM ITS PARTITION
   1381                                                .ENDC
   1382
   1383                                      ;+
   1384                                      ; DEALLOCATE DECNET HOME BLOCK
   1385                                      ;-
   1386 004360  042767  000000G 000000G                BIC     #FE.CEX,$FMASK   ; COMM EXEC NOT IN MEMORY
   1387 004366  016700  000000G                        MOV     $DECPT,R0        ; GET HOME BLOCK ADDRESS
   1388 004372  001404                                 BEQ     61$              ; IF NONE - BRANCH
   1389 004374  012701  000104                         MOV     #D$END,R1        ; GET SIZE
   1390 004400                                         CALL    $DEA16           ; DEALLOCATE BLOCK FROM DSR
   1391
   1392
   1393 004404  005067  000000G               61$:     CLR     $BFR             ; CLEAR ADDRESS OF POWERFAIL ROUTINE
   1394 004410                                         PUTRC   .NETPF,#2        ; ...
   1395
   1396                                                .IF     DF,R$$MPL
   1397                                                MOV     .MPRO,R0         ; GET NUMBER OF PROCESSORS
   1398                                                BLE     65$              ; BR IF SINGLE PROCESSOR SYSTEM
   1399                                      60$:      MOV     R0,R2            ; COPY PROCESSOR NUMBER
   1400                                                MOV     R0,-(SP)         ; SAVE PROCESSOR NUMBER
   1401                                                DEC     R2               ; MAKE PROCESSOR NUMBER A ZERO OFFSET
   1402                                                ASL     R2               ; MAKE IT INTO A WORD INDEX
   1403                                                ADD     .K6TAB,R2        ; POINT INTO CPU AREA TABLE
   1404                                                SWTCHI  #$CPBIA          ; SET TO READ INTO CPU AREA BIAS ARGUMENT
   1405                                                GETRV   R2,#2            ; GET BIAS OF CPU'S AREA
   1406                                                SWTCHI  #UMRPTR          ; SET TO READ INTO THE UMR POINTER
   1407                                                MOV     $CPBIA,$RBLCK    ; RESET THE APR5 BIAS
   1408                                                MOV     .UMRPT,R2        ; GET THE UMR BLOCK POINTER OFFSET
   1409                                                GETAD   R2,#2            ; GET THE UMR BLOCK POINTER FOR THE CPU
   1410                                                SWTCHI                   ; RESET TO STANDARD INPUT BUFFER
```

```
                                                                 M  1
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 5
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE               REFERENCES

OP$LOC  = 000001             #5-63       #5-63
OP$LST  = 000002             #5-63
OP$LTM  = 000001             #5-63
OP$LVL  = 000010             #5-63
OP$MAC  = 000010             #5-63
OP$MCO  = 000002             #5-63
OP$MDE  = 000400             #5-63
OP$MLI  = 000004             #5-63
OP$MON  = 000100             #5-63
OP$MST  = 000020             #5-63
OP$NAM  = 000004             #5-63       #5-63       #5-63
OP$NET  = 004000             #5-63
OP$NLI  = 000002             #5-63        14-396      14-404
OP$NOD  = 000020             #5-63
OP$OUT  = 002000             #5-63
OP$OWN  = 000020             #5-63
OP$PAR  = 000010             #5-63
OP$PRI  = 000010             #5-63
OP$RET  = 010000             #5-63
OP$RPA  = 000010             #5-63        14-396
OP$SCO  = 000001             #5-63
OP$SEG  = 000400             #5-63
OP$SER  = 000400             #5-63
OP$SIN  = 000010             #5-63
OP$STA  = 000004             #5-63
OP$TPA  = 000020             #5-63        14-396
OP$TRI  = 000040             #5-63
OP$UCS  = 000020             #5-63
OP$USE  = 000002             #5-63
OP$VEC  = 000040             #5-63
OP$VER  = 000040             #5-63        14-396
OP$XAC  = 000001             #5-63
OP$XPR  = 000002             #5-63
OP$X5S  = 000004             #5-63
OP$X9S  = 000010             #5-63
PBITS     000356 R           #7-192       17-534
PDVA      000474 R             6-118      6-128      #15-467
PDVX      000246 R           #7-178       14-410     *19-597      24-901      *28-1062     29-1093
PSILD     000216 R             6-141      6-147      #13-372
QF$ACT  = 000001             #5-63
QF$ALL  = 000002             #5-63
QF$KNO  = 000004             #5-63
QF$LOO  = 000010             #5-63
REP7      003344 R             8-209      8-210       8-211       8-212       8-213       8-214       8-215       8-216       8-217
                               8-218      8-219       8-220       8-221       8-222       8-226       8-227       8-228       8-229
                               8-230      8-231       8-232       8-233       8-234       8-235       8-236       8-237      #31-1176
                               32-1193    33-1226     34-1251
REP7A     003346 R             8-223     #32-1192
REP7B     003456 R             8-225     #34-1250
RNTI      000162 R             6-161     #11-337
RQB     = ******     GX      *14-441      18-568
R$$EIS  = ******               20-685

                                                                 N  1
VRQB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55      PAGE 6
```

```
VSET        CREATED BY  MACRO  ON 29-JUN-85 AT 02:59      PAGE 2
MACRO CROSS REFERENCE                                      CREF   04.00

MACRO NAME        REFERENCES

CALL               5-83        5-103
PUTAD             #5-72        5-103
RAND              #5-103       5-103
RETURN             5-114
SWTCHI            #5-72        5-113
SWTCHO            #5-72        5-102       5-112
```

VS
SE

FILEID**VSTB

VS

```
    463                                 ;
    464                                 ; SET MODULE
    465                                 ;
    466 000000                                  STATE$  SEMO,&P$SEMO
    467 000000                                   TRAN$   $LAMDA,         SEMOX5,     $IFX5S
    468                                 ;
    469                                 ;   SET MODULE X25-SERVER OPTIONS
    470                                 ;
    471 000000                                  STATE$  SEMOX5
    472 000000                                   TRAN$   !OPX5ST,        SEMOX5,     ,           OP$MST,         $OPTON
    473 000000                                   TRAN$   $LAMDA,         DONE
    474                                 ;
    475                                 ; SET OBJECT
    476                                 ;
    477 000000                                  STATE$  SEOB,&P$SEOB
    478 000000                                   TRAN$   !SEOBOP
    479                                 ;
    480 000000                                  STATE$  SEOB1
    481 000000                                   TRAN$   $EOS,           DONE
    482 000000                                   TRAN$   !SEOBOP,        SEOB1
    483                                 ;
    484                                 ;     SET OBJECT OPTIONS
    485                                 ;
    486 000000                                  STATE$  SEOBOP
    487 000000                                   TRAN$   !OPOBNA,        $EXIT,      ,           OP$NAM,         $OPTON
    488 000000                                   TRAN$   !OPCOP,         $EXIT,      ,           OP$COP,         $OPTON
    489 000000                                   TRAN$   !OPUSE,         $EXIT,      ,           OP$USE,         $OPTON
    490 000000                                   TRAN$   !OPOBVE,        $EXIT,      ,           OP$LVL,         $OPTON
    491                                 ;
    492                                 ; SET PROCESS
    493                                 ;
    494 000000                                  STATE$  SEPR,&P$SEPR
    495 000000                                   TRAN$   &K$ALL, DONE
    496 000000                                   TRAN$   !SEPROP
    497                                 ;
    498 000000                                  STATE$  SEPR1
    499 000000                                   TRAN$   $EOS,           DONE
    500 000000                                   TRAN$   !SEPROP,        SEPR1
    501                                 ;
    502                                 ;     SET PROCESS OPTIONS
    503                                 ;
    504 000000                                  STATE$  SEPROP
    505 000000                                   TRAN$   !OPLOC,         $EXIT,      ,           OP$LOC,         $OPTON
    506 000000                                   TRAN$   !OPMXCO,        $EXIT,      ,           OP$MCO,         $OPTON
    507 000000                                   TRAN$   !OPMXLI,        $EXIT,      ,           OP$MLI,         $OPTON
    508 000000                                   TRAN$   !OPPAR,         $EXIT,      ,           OP$PAR,         $OPTON
    509                                 ;
    510                                 ; SET SYSTEM
    511                                 ;
    512 000000                                  STATE$  SESY
    513 000000                                   TRAN$   &K$ALL,         SESY
    514 000000                                   TRAN$   &K$ECH,         SESY,       ,           VF.ECH,         $VFLAG
    515 000000                                   TRAN$   $LAMDA,         DONE
```

```
   1127                                      ;
   1128                                      ; LOGGING STATE
   1129                                      ;
   1130 000000                                      STATE$  OPLGST
   1131 000000                                       TRAN$  &K$STAE
   1132 000000                                      STATE$   ,&P$LOST
   1133 000000                                       TRAN$  &K$OFF,          $EXIT
   1134 000000                                       TRAN$  &K$ON,           $EXIT,        ,           VF.ON,          $VFLAG
   1135                                      ;
   1136                                      ; MODULE X25-SERVER STATE
   1137                                      ;
   1138 000000                                      STATE$  OPX5ST
   1139 000000                                       TRAN$  &K$STAE
   1140 000000                                      STATE$   ,&P$X5ST
   1141 000000                                       TRAN$  &K$OFF,          $EXIT
   1142 000000                                       TRAN$  &K$ON,           $EXIT,        ,           VF.ON,          $VFLAG
   1143                                      ;
   1144                                      ; MULTIPOINT ACTIVE
   1145                                      ;
   1146 000000                                      STATE$  OPMACT
   1147 000000                                       TRAN$  &K$MUL
   1148 000000                                      STATE$   ,&P$MUL
   1149 000000                                       TRAN$  &K$ACT
   1150 000000                                      STATE$   ,&P$ACT
   1151 000000                                       TRAN$  $DNUMB,          $EXIT,        $MACT
   1152                                      ;
   1153                                      ; MAXIMUM CONTROLLERS
   1154                                      ;
   1155 000000                                      STATE$  OPMXCO
   1156 000000                                       TRAN$  &K$MAX
   1157 000000                                      STATE$   ,&P$MAX
   1158 000000                                       TRAN$  &K$CONS
   1159 000000                                      STATE$   ,&P$MXCO
   1160 000000                                       TRAN$  $DNUMB,          $EXIT,        $MXCO,      LS.OPT,         RQBST
   1161                                      ;
   1162                                      ; MAXIMUM LINES
   1163                                      ;
   1164 000000                                      STATE$  OPMXLI
   1165 000000                                       TRAN$  &K$MAX
   1166 000000                                      STATE$   ,&P$MAX
   1167 000000                                       TRAN$  &K$LINS
   1168 000000                                      STATE$   ,&P$MXLI
   1169 000000                                       TRAN$  $DNUMB,          $EXIT,        $MXLI,      LS.OPT,         RQBST
   1170                                      ;
   1171                                      ; MULTIPOINT DEAD
   1172                                      ;
   1173 000000                                      STATE$  OPMDEA
   1174 000000                                       TRAN$  &K$MUL
   1175 000000                                      STATE$   ,&P$MUL
   1176 000000                                       TRAN$  &K$DEA
   1177 000000                                      STATE$   ,&P$DEA
   1178 000000                                       TRAN$  $DNUMB,          $EXIT,        $MDEA
   1179                                      ;
   1180                                      ; NODE ADDRESS
   1181                                      ;
   1182 000000                                      STATE$  OPADD
   1183 000000                                       TRAN$    &K$ADD
```

```
                                ;
                                ; CHECK AHEAD IN INPUT LINE FOR SLASH, UNDERSCORE OR DOUBLE COLON TO END
                                ; DESTINATION STRING
                                ;
000062                          DSTR.E:
                                ;       IF INPUT LENGTH <= 0 OR NEXT INPUT CHARACTER <> / OR _ OR ::
000062  005703                            TST     R3                    ;CHECK INPUT LENGTH
000064  003415                            BLE     10$                   ;TRUE IF INPUT LENGTH <= 0
000066  121427  000057                    CMPB    (R4),#<'/>            ;COMPARE NEXT CHARACTER TO ASCII /
000072  001414                            BEQ     20$                   ;FALSE IF CHARACTER = ASCII /
000074  121427  000137                    CMPB    (R4),#<'_>            ;COMPARE NEXT CHARACTER TO ASCII _
000100  001411                            BEQ     20$                   ;FALSE IF CHARACTER = ASCII _
000102  121427  000072                    CMPB    (R4),#<':>            ;COMPARE NEXT CHARACTER TO ASCII :
000106  001004                            BNE     10$                   ;TRUE IF CHARACTER <> ASCII :
000110  126427  000001  000072            CMPB    1(R4),#<':>           ;COMPARE AFTER NEXT CHARACTER TO ASCII :
000116  001402                            BEQ     20$                   ;FALSE IF TWO NEXT CHARACTERS = ASCII :
000120                          10$:;   THEN
000120  062716  000002                    ADD     #2,(SP)               ;REJECT TRANSITION
000124                          20$:;   ENDIF

000124                                  RETURN
```

```
VSTB       CREATED BY  MACRO  ON 29-JUN-85 AT 03:03     PAGE 8
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE              REFERENCES

$XTRI   = ******  G          5-99
$XUNT   = ******  G          5-101
$$$FLG  = 177777            #6-108
$$$KEY  = 177777            #6-108
.PFLAG  = ******  G          5-85       *17-1596    *17-1603    *18-1620    *18-1624
```

```
VSTB       CREATED BY  MACRO  ON 29-JUN-85 AT 03:03     PAGE 9
```

```
Work  file writes: 0
Size of work file: 12727 Words  ( 50 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:09.58
SY:VSYM.V2,[132,134]VSYM/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VSYM
```

N 7

```
                                        .SBTTL  $TMP - TEMPLATE FILE PROCESSOR
                                ;+
                                ; $TMP - TEMPLATE FILE PROCESSOR
                                ;
                                ; THIS ROUTINE IS CALLED TO PROCESS THE TEMPLATE FILE AND TO LOAD THE
                                ; DATA BASE INTO CORE.
                                ;
                                ; INPUTS:
                                ;       $BIAS=PROCESS RELOCATION BIAS
                                ;       $$CSR=CSR ADDRESS
                                ;       $$PRI=INTERRUPT PRIORITY
                                ;       $$VECT=INTERRUPT VECTOR
                                ;
                                ; OUTPUTS:
                                ;       C-BIT=SUCCESS/FAILURE
                                ;-
000000                          $TMP::
000000  005067  000014'                 CLR     $SADDR          ; NO LINE TABLE YET
000004                                  CALL    $TM.IN          ; INITIALIZE COUNTERS, OPEN PRIMARY TEMPLATE
000010  103420                          BCS     10$             ; IF CS, OPEN FAILURE OR NOTHING TO DO
000012                                  CALL    $TPARS          ; LOAD TPARS TABLES
000016                                  CALL    $TMFIL          ; PROCESS TEMPLATE FILE

000022                                  CALL    $TMCHK          ; CHECK FOR DDM INTERRUPT LINKAGE ERROR
000026                                  CALL    $TMALT          ; ALLOCATE DATA BASE (VALT)
000032                                  CALL    $VEXP           ; EXPAND BINARY INFORMATION (TMPEXP..VEX2)

000036                                  CALL    $TMVCT          ; SETUP DDM VECTOR(S) (VVCT)

000042                                  CALL    $TMLTA          ; STORE LINE TABLE ADDRESS (VVCT)
000046                                  CALL    IMGRES          ; WRITE IMAGES TO SYSTEM IMAGE FILE
000052  006167  000000'         10$:    ROL     $FLAG1          ; INTO THE C-BIT
000056                                  SWTCHO                  ; POINT BACK AT DEFAULT BUFFER
000064                                  RETURN

                                ;
                                ; PROCESS TEMPLATE FILE
                                ;
000066                          $TMFIL: CALL    $SAVAL          ; SAVE ALL REGISTERS
000072  016746  000002'                 MOV     $FLAG2,-(SP)    ; SAVE $FLAG2
000076  062767  000002  000036'         ADD     #2,$SZPTR       ; ADJUST SIZE POINTER FOR NEW DEPTH
000104  016777  000012' 000036'         MOV     $SIZE,@$SZPTR   ; SAVE CURRENT VALUE OF $SIZE
000112  005067  000002'         10$:    CLR     $FLAG2          ; CLEAR LINE FLAGS
000116  005067  000276'                 CLR     $VALUE          ; CLEAR EXPRESSION VALUE
000122                                  CALL    TM.RD           ; READ 1 LINE FROM TEMPLATE FILE
000126  103426                          BCS     30$             ; IF CS, END OF FILE
000130  032767  000000G 000000G         BIT     #FL.ECH,$FLAGS  ; IS ECHO ENABLED?
000136  001402                          BEQ     15$             ; IF EQ, NO
000140                                  CALL    $ELINE          ; PRINT TEMPLATE LINE
000144  005001                  15$:    CLR     R1              ; PARSE THE INPUT LINE
000146  012702  000000G                 MOV     #$TMPKW,R2
000152  012705  000000G                 MOV     #$TMPST,R5
000156                                  CALL    .TPARS
000162  103002                          BCC     20$             ; IF CC, NO PROBLEM
000164                                  CALL    ERR$9           ; SYNTAX ERROR
000170  032767  000002  000002' 20$:    BIT     #F2.END,$FLAG2  ; ".END" SEEN YET?
000176  001745                          BEQ     10$             ; IF EQ, NO - GET NEXT LINE
```

```
                                                                  M  9
VTMP - VNP TEMPLATE PROCESSING  MACRO V05.03b Saturday 29-Jun-85 03:08  Page 18-1
Symbol table

A$$CHK= 000000        F1.MXT= 001000 G      P$$P45= 000000        $ERR21  000766R   002 $RSIZE= ****** GX
A$$CPS= 000000        F1.STE= 002000 G      P$$WRD= 000000        $ERR79  001014R   002 $SADDR  000014RG  002
A$$PRI= 000000        F1.UMR= 004000 G      Q$$OPT= 000010        $ERR82  001042R   002 $SAVAL= ****** GX
A$$TRP= 000000        F1.VSE= 010000 G      R$$DER= 000000        $ERR83  001110R   002 $SAVVR= ****** GX
BUFCK   000560R       F2.BYT= 000001 G      R$$K11= 000001        $EVEN   001312RG      $SIZE   000012RG  002
C$$CKP= 000000        F2.END= 000002 G      R$$SND= 000000        $FILEN  000302RG  002 $SZPTR  000036RG  002
C$$ORE= 000400        F2.FIL= 000004 G      R$$11M= 000000        $FILEX= ****** GX     $TADDR  000016RG  002
C$$RSH= 177564        F2.LBR= 000010 G      SYSFDB= ****** GX     $FLAGS= ****** GX     $TERR   001470RG
D$$BUG= 177514        F2.NEG= 000020 G      S$$WRG= 000000        $FLAG1  000000RG  002 $TIOUT= ****** GX
D$$ISK= 000000        F2.SUB= 000040 G      S$$YSZ= 007600        $FLAG2  000002RG  002 $TMALT= ****** GX
D$$L11= 000001        G$$TPP= 000000        TMPBF   000352R   002 $FNR50  000340RG  002 $TMCHK= ****** GX
D$$YNC= 000000        G$$TSS= 000000        TMPSZ   000556R   002 $GCLEP= ****** GX     $TMFIL  000066R
D$$YNM= 000000        G$$TTK= 000000        TM.RD   001056R       $ICBAD  000272RG  002 $TMFIX  000220RG
EPFX    000713R   002 G$$WRD= 000000        T$$KMG= 000000        $ICBSZ  000274RG  002 $TMLTA= ****** GX
ERRBUF  000565R   002 IE.EOF= ****** GX     T$$MIN= 000000        $LBADR  000346RG  002 $TMP    000000RG
ERRMSG  000560R   002 IE.RBG= ****** GX     V$$CTR= 001000        $LBIAS  000350RG  002 $TMPBF= ****** GX
ERR$10= ****** GX     IMGRES  000240R       WPFX    000676R   002 $LDA    000060RG  002 $TMPKW= ****** GX
ERR$11= ****** GX     I$$RAR= 000000        X$$DBT= 000000        $LEFT   000010RG  002 $ MPND= ****** GX
ERR$12= ****** GX     I$$RDN= 000000        $ABFI = ****** GX     $LFLD   000322RG  002 $  PST= ****** GX
ERR$9 = ****** GX     K$$CNT= 177546        $ADDR   000020RG  002 $LINK   000026RG  002 $TMVCT= ****** GX
ERR1    001050R       K$$CSR= 177546        $ADRX = ****** GX     $MPLD   000320RG  002 $TM.IN= ****** GX
E$$XPR= 000000        K$$LDC= 000000        $ADRZ = ****** GX     $MXADD  000022RG  002 $TPARS= ****** GX
FATAL = ****** GX     K$$TPS= 000074        $BFALL  000324RG  002 $MXLEN  000032RG  002 $TWARN  001444RG
FL.DDM= ****** GX     LD$LP = 000000        $BFBAS  000336RG  002 $MXSIZ  000024RG  002 $TWE    001460RG
FL.DLC= ****** GX     L$$ASG= 000000        $BFLBN  000326RG  002 $NALL   000034RG  002 $UBIAS  000054RG  002
FL.ECH= ****** GX     L$$DRV= 000000        $BFPTR  000332RG  002 $NLIB   000035RG  002 $UDA    000056RG  002
FL.LLC= ****** GX     L$$P11= 000001        $BFR  = ****** GX     $NTLLB  000162RG  002 $VALUE  000276RG  002
FMT2    001154R   002 L$$11R= 000000        $BFXFR  000334RG  002 $NTLUB  000062RG  002 $VECT1  000262RG  002
FM.2  = 000000        M$$CRB= 000124        $BIAS = ****** GX     $NVECT  000270RG  002 $VECT2  000264RG  002
F$$LVL= 000001        M$$CRX= 000000        $BIN    000004RG  002 $OFF    000030RG  002 $VECT3  000266RG  002
F.NRBD= ****** GX     M$$FCS= 000000        $BUFCK  000316RG      $PTR    000006RG  002 $VEXP = ****** GX
F.RSIZ= ****** GX     M$$MGE= 000000        $CAPR   000316RG  002 $PUTAD= ****** GX     $VFDDM  001344RG
F1.BBO= 000001 G      M$$NET= 000000        $CLBUF= ****** GX     $RADDR= ****** GX     $VFDLC  001364RG
F1.ERR= 100000 G      M$$OVR= 000000        $CLDEQ= ****** GX     $RBLCK= ****** GX     $VFLV1  001416RG
F1.INT= 000002 G      N$$ACC= 000001        $CLIOS= ****** GX     $REP.C= ****** GX     $VFSPC  001404RG
F1.IN1= 000004 G      N$$BUF= 000001        $CLSIZ= ****** GX     $RQ1    001226RG      $$    = 177777
F1.IN2= 000010 G      N$$LDV  000001        $CVAL   000314RG  002 $RQ2    001220RG      $$$   = 000630R
F1.IN3= 000020 G      N$$MC   000001        $DIFF   000300RG  002 $RQ3    001212RG      $$$$  = 001050R
F1.LFI= 000040 G      N$$ML   000001        $ELINE  001154RG      $RQ4    001204RG      .ABFI   000230RG
F1.LFL= 000100 G      N$$MOV= 000010        $EROUT= ****** GX     $RQ5    001176RG      .BYTE = ****** GX
F1.MPL= 000200 G      N$$NCT= 000001        $ERR13  000720R   002 $RQ7    001170RG      .TPARS= ****** GX
F1.MPT= 000400 G      N$$PEM= 000001

. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        001576    001   (RW,I,LCL,REL,CON)
DATA    001174    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10881 Words  ( 43 Pages)
Size of core pool: 14440 Words  (  5 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:22.21
```

```
                                                                  N  9
```

```
 161                                           .IF DF   R$$MPL
 162
 163                                     4$:   MOVB     K.VCT-K.PRM+$BFR,R1 ; GET VECTOR ADDRESS
 164                                           .IFF
 165 000210  116701  000001G            4$:   MOVB     K.VCT+$BFR,R1    ; GET THE VECTOR ADDRESS
 166
 167                                           .ENDC
 168 000214  006301                           ASL      R1               ; CONVERT TO REALL ADDRESS
 169 000216  006301                           ASL      R1               ;     ...
 170 000220                                   CALL     FNDADD           ; FIND LINE TABLE ADDRESS
 171 000224  020167  000000G                  CMP      R1,$$VECT        ; IS THIS THE SAME VECTOR ADDRESS ?
 172 000230  001403                           BEQ      10$              ; IF NE, NO .. WARN USER
 173 000232                                   EMSG$    WZ               ; WARNING - MUX UPDATE ONLY
 174 000240  005000                     10$:  CLR      R0               ; SETUP ASCIZ FILE SPEC
 175 000242  005001                           CLR      R1               ; FOR [131,5X]XXX.DAT
 176 000244  012702  000000'                  MOV      #TMPEXT,R2       ;     ...
 177 000250                                   CALL     $AZFS            ;     ...
 178 000254                                   CALL     $CLQUE           ; TRY TO OPEN PRIMARY TEMPLATE
 179 000260  105777  000000G                  TSTB     @$CLIOS          ; SUCCESS? (CLEARS C-BIT)
 180 000264  100401                           BMI      101$             ; IF MI, NO
 181 000266                           RTSPC:  RETURN
 182
 183                                     ;
 184                                     ; ERROR CONDITIONS
 185                                     ;
 186 000270  032767  000000G 000000G 101$:    BIT      #FL.LLC,$FLAGS   ; IS THIS AN LLC?
 187 000276  001405                           BEQ      102$             ; IF EQ, NO
 188 000300  016700  000000G                  MOV      $PDVA,R0         ; GET PDV ADDRESS
 189 000304  012760  177777  000000G          MOV      #-1,Z.DAT(R0)    ; MARK DATA BASE AS EMPTY
 190 000312                           102$:   EMSG$R   8                ; OPEN FAILURE
 191                                           .DSABL   LSB
```

```
    229 000702  001134'                                     .WORD   GETRC
    230 000704     107      101                             .ASCII  /GA/
    231 000706  001216'                                     .WORD   GETAD
    232 000710     120      122                             .ASCII  /PR/
    233 000712  001436'                                     .WORD   PUTRC
    234 000714     120      101                             .ASCII  /PA/
    235 000716  001326'                                     .WORD   PUTAD
    236 000720     104      102                             .ASCII  /DB/
    237 000722  001520'                                     .WORD   DUMPB
    238 000724     104      127                             .ASCII  /DW/
    239 000726  002054'                                     .WORD   DUMPW
    240 000730     115      117                             .ASCII  /MO/
    241 000732  002412'                                     .WORD   MODIFY
    242 000734     101      066                             .ASCII  /A6/
    243 000736  002710'                                     .WORD   ALL16
    244 000740     101      062                             .ASCII  /A2/
    245 000742  003264'                                     .WORD   ALL22
    246 000744     104      066                             .ASCII  /D6/
    247 000746  003154'                                     .WORD   DEA16
    248 000750     104      062                             .ASCII  /D2/
    249 000752  003530'                                     .WORD   DEA22
    250 000754     117      104                             .ASCII  /OD/
    251 000756  000764'                                     .WORD   ODT
    252 000760  000000                                      .WORD   0
    253 000762  000000                                      .WORD   0
    254
    255                                             ; SUBROUTINES
    256
    257 000764                                      ODT:    SVTK$S  #SSTTBL,#3      ; 3 WORD SST TABLE
    258 001002  000003                                      BPT                     ; TRAP
    259 001004                                              SVTK$S                  ; DEALLOCATE TABLE
    260 001016                                              RETURN
    261
    262 001020                                      BPT:    TYPES   <VNP -- TASK NOT BUILT WITH ODT>
    263 001050  000002                                      RTI                     ; RETURN FROM ODT TRAP
    264                                             ;
    265                                             ; GET 16BIT ADDRESS WITH EXEC CHECK
    266                                             ;
    267 001052                                      GETRV:  GETNUM  <VIRTUAL ADDRESS>,$RADDR,X101
    268 001100                                              GETNUM  <NUMBER OF BYTES>,$RSIZE,X101
    269 001126                                              GETRV
    270 001132                                      X101:   RETURN
    271                                             ;
    272                                             ; GET 16BIT VIRTUAL ADDRESS - NO EXEC CHECK
    273                                             ;
    274 001134                                      GETRC:  GETNUM  <VIRTUAL ADDRESS>,$RADDR,X102
    275 001162                                              GETNUM  <NUMBER OF BYTES>,$RSIZE,X102
    276 001210                                              GETRC
    277 001214                                      X102:   RETURN
    278                                             ;
    279                                             ; GET 22BIT ADDRESS
    280                                             ;
    281 001216                                      GETAD:  GETNUM  <BIAS>,$RBLCK,X103
    282 001244                                              GETNUM  <VIRTUAL ADDRESS>,$RADDR,X103
    283 001272                                              GETNUM  <NUMBER OF BYTES>,$RSIZE,X103
    284 001320                                              GETAD
    285 001324                                      X103:   RETURN
```

```
                                  ;+
                                  ; $ASUMR - ASSIGN UNIBUS MAPPING REGISTERS
                                  ;
                                  ; THIS ROUTINE IS CALLED TO SCAN THE LIST OF CURRENTLY ALLOCATED UMR'S
                                  ; ($UMRHD), ALLOCATE CONTIGUOUS UMR'S SPECIFIED BY THE CALLER (M.UMRN),
                                  ; AND LINK THIS UMR BLOCK TO THE OTHERS.
                                  ;
                                  ; INPUTS:
                                  ;       R0=UMR BLOCK REAL MEMORY ADDRESS
                                  ;       R1=UMR BLOCK ADDRESS (IN VNP TASK SPACE)
                                  ;       R2=PROCESSOR NUMBER IF MULTIPROCESSOR
                                  ;       .UMRHD=VALUE OF $UMRHD FROM RSX11M.STB FILE
                                  ;
                                  ;       ON MULTIPROCESSOR RSX-11M-PLUS SYSTEMS, .CPBIA MUST BE SET TO
                                  ;       THE BIAS THE CPU'S DATA AREA.
                                  ;
                                  ; OUTPUTS:
                                  ;       C-BIT=SUCCESS/FAILURE
                                  ;       ALL REGISTERS ARE PRESERVED
                                  ;-
000000                            $ASUMR::CALL    $SAVAL          ; SAVE ALL REGISTERS
000004                                    CALL    GFIRST          ; GET FIRST UMR BLOCK(S)
000010  000402                            BR      20$

000012                            10$:    CALL    GNEXT           ; GET NEXT UMR BLOCK
000016  103456                    20$:    BCS     50$             ; IF CS, FAILURE
000020  016704  000002'                   MOV     PREV+M.UMRA,R4  ; CALCULATE FIRST UMR
000024  066704  000004'                   ADD     PREV+M.UMRN,R4  ; NOT MAPPED BY THIS BLOCK
000030  012746  170374                    MOV     #UBMPR+<31.*4>,-(SP) ; POINT AT FIRST NON-EXISTANT UMR
000034  005703                            TST     R3              ; IS THERE A NEXT BLOCK?
000036  001402                            BEQ     30$             ; IF EQ, NO
000040  016716  000016'                   MOV     NEXT+M.UMRA,(SP) ; ELSE USE FIRST UMR FROM NEXT BLOCK
000044  160416                    30$:    SUB     R4,(SP)         ; GIVING FREE UMR COUNT
000046  022661  000004                    CMP     (SP)+,M.UMRN(R1) ; ENOUGH FOR OUR USE?
000052  103757                            BLO     10$             ; IF LO, NO
000054  010461  000002                    MOV     R4,M.UMRA(R1)   ; STASH UMR ADDRESS IN USER'S BLOCK
000060  010067  000000'                   MOV     R0,PREV         ; LINK PREVIOUS BLOCK TO OURS
000064  062767  000002  000000'           ADD     #2,PREV         ;       ...
000072  010311                            MOV     R3,(R1)         ; LINK OUR BLOCK TO NEXT
000074  001402                            BEQ     40$             ;       ...
000076  062711  000002                    ADD     #2,(R1)         ;       ...
000102  010405                    40$:    MOV     R4,R5           ; CONVERT UMR ADDRESS TO 22-BIT ADDRESS
000104  142705  170234                    BICB    #UBMPR!34,R5    ;       ...
000110  040504                            BIC     R5,R4           ;       ...
000112  006205                            ASR     R5              ;       ...
000114  106304                            ASLB    R4              ;       ...
000116  006304                            ASL     R4              ;       ...
000120  006304                            ASL     R4              ;       ...
000122  000304                            SWAB    R4              ;       ...
000124  010461  000006                    MOV     R4,M.UMVL(R1)   ; STUFF 22-BIT ADDRESS INTO USER'S BLOCK
000130  110561  000010                    MOVB    R5,M.UMVH(R1)   ;       ...
000134                                    SWTCHO  #PREV           ; CHANGE DEFAULT BUFFER

000142                                    PUTRC   R2              ; RE-WRITE PREVIOUS BLOCK
000152  000241                    47$:    CLC                     ; INDICATE SUCCESS
000154                            50$:    SWTCHI                  ; RESET DEFAULT BUFFERS
000162                                    SWTCHO                  ;       ...
```

```
                                        ;+
                                        ; LINK TO NEXT BLOCK
                                        ;-

 000166  004267  000000G                LINK:   JSR     R2,$SAVVR               ; SAVE VOLATILE REGISTERS
 000172  010004                                 MOV     R0,R4                   ; COPY POINTER
 000174  016701  000000G                        MOV     $UNBSZ,R1               ; GET $UNLOAD BLOCK SIZE
 000200                                         SWTCHI                          ; RESET DEFAULT BUFFER
 000206                                         SWTCHO                          ; ...
 000214                                         CALL    $ALN18                  ; ALLOCATE BLOCK
 000220  103445                                 BCS     101$                    ; RESOURCE ERROR
 000222                                         SWTCHO  #$UNBLK                 ; SET DEFAULT BUFFER TO BLOCK IMAGE
 000230                                         SWTCHI  #$UNBLK                 ; ...
 000236  016777  000000G 000000G                MOV     $UNBAD,@$ADRX           ; SET ADDRESS OF CURRENT BLOCK
 000244  010014                                 MOV     R0,(R4)                 ; STORE LINK VALUE
 000246  020427  000000G                        CMP     R4,#$UNBAD              ; IS THIS LISTHEAD ?
 000252  001007                                 BNE     5$                      ; YES, HOMEBLOCK REWRITTEN IN VXIT
 000254  010067  000000G                        MOV     R0,$UNBLK               ; LOAD LINK WORD
 000260                                         PUTAD   ,#64.                   ; WRITE PREVIOUS BLOCK
 000272  012704  000000G                5$:     MOV     #$UNBLK,R4              ; ZERO COPY OF NEWLY ALLOCATED BLOCK
 000276  012703  000040                         MOV     #32.,R3                 ; ...
 000302  005024                         11$:    CLR     (R4)+                   ; ...
 000304  005303                                 DEC     R3                      ; ...
 000306  001375                                 BNE     11$                     ; ...
 000310  010077  000000G                        MOV     R0,@$ADRX               ; SET UP VIRTUAL ADDRESS
 000314  012767  000100   000000G               MOV     #64.,$RSIZE             ;  AND SIZE
 000322                                         PUTAD                           ; WRITE ZEROED BLOCK TO SYS IMAGE
 000326  010067  000000G                        MOV     R0,$UNBAD               ; SAVE CURRENT BLOCK'S VA
 000332                                         RETURN

                                        ; ERROR MESSAGE

 000334                                 101$:   ERRPT$  ER1,$EROUT,FATAL        ; NTPOOL TOO SMALL
 000350                                 102$:   ERRPT$  ER2,$EROUT,FATAL        ; LOGIC ERROR IN BLOCK MANIPULATION
 000364                                         ERMSG$  <Network buffer allocation failure ( 32. words ).>
 000444                                         ERBLK$  ER1
 000450                                         ERMSG$  <Logic error manipulating accounting blocks.>
 000523                                         ERBLK$  ER2

                                                .DSABL  LSB

         000001                                 .END
```

```
000366                                  CALL    $XPAR           ; RELEASE PROCESS FROM PARTITION
000372  103452                          BCS     111$            ; IF CS, VERY BAD

                                        .IF DF  R$$MPL
                                        CALL    DEVEC           ; DEALLOCATE VECTOR BLOCK
                                        .ENDC

000374  005064  000000G                 CLR     Z.PCB(R4)       ; MARK PROCESS NON-RESIDENT
000400  005064  000000G                 CLR     Z.DSP(R4)       ; ZERO DISPATCH TABLE ADDRESS
000404  005064  000002G                 CLR     Z.DSP+2(R4)     ; ZERO RELOCATION BIAS

000410  132764  000000G 000000G 10$:    BITB    #ZF.LLC,Z.FLG(R4) ; IS THIS AN LLC?
000416  001435                          BEQ     20$             ; IF EQ, NO
000420  016403  000000G                 MOV     Z.DAT(R4),R3    ; GET LINE TABLE ADDRESS
000424  001432                          BEQ     20$             ; IF NO LINE TABLE, INIT FAILURE
000426  022703  177777                  CMP     #-1,R3          ; IS THIS AN EMPTY LLC LINE TABLE?
000432  001425                          BEQ     19$             ; IF EQ, YES
000434  005067  000202'                 CLR     ..PDV           ; DISCARD ANY PARTITION LINE TABLES

000440  022764  114224  000000G         CMP     #^RXPT,Z.NAM(R4) ; Is this XPT?
000446  001003                          BNE     15$             ; If not, skip
000450                                  CALL    DEAXPT          ; If so, deallocate XPT data structures
000454  000406                          BR      18$             ; continue

000456  022764  017704  000000G 15$:    CMP     #^RECL,Z.NAM(R4) ; Is this ECL?
000464  001002                          BNE     18$             ; If not, skip
000466                                  CALL    DEAECL          ; If so, deallocate ECL Node counters

000472  010446                  18$:    MOV     R4,-(SP)        ; SAVE PDV ADDRESS
000474                                  CALL    GETIND          ; GET PDV INDEX
000500                                  CALL    XLT16           ; DE-ALLOCATE LINE TABLE
000504  012604                          MOV     (SP)+,R4        ; RESTORE PDV ADDRESS
000506  005064  000000G         19$:    CLR     Z.DAT(R4)       ; ZERO LINE TABLE ADDRESS

000512  000261                  20$:    SEC                     ; RETURN WITH C-BIT SET
000514                                  RETURN

                                ;
                                ; ERROR CONDITIONS
                                ;
000516                          101$:   CRASH                   ; NETLDR LOGIC ERROR
000520                          111$:   CRASH                   ; SOMEBODY UNLINKED OUR SUB-PCB!!!
                                        .DSABL  LSB
```

```
    869 002340  001007                                BNE     30$              ; IF NE, NO
    870 002342  005201                                INC     R1               ; TALLY IT
    871 002344  126365  000000G 000000G               CMPB    L.CTL(R3),L.CTL(R5) ; SAME CONTROLLER AS OURS?
    872 002352  001002                                BNE     30$              ; IF NE, NO
    873 002354  005267  000004'                       INC     UNITS            ; TALLY IT
    874 002360  005304                        30$:    DEC     R4
    875 002362  001351                                BNE     10$              ; LOOP <SLT COUNT> TIMES
    876 002364  010067  000120'                       MOV     R0,DLCAC         ; SETUP ACTIVITY COUNTS
    877 002370  010167  000122'                       MOV     R1,DDMAC         ;      ...
    878 002374  016702  000004'                       MOV     UNITS,R2         ; RETURN OTHER READY LINES ON THIS CONTROLLER
    879 002400                                        RETURN
    880
```

```
                                             MOV     UMRPTR,R0          ; COPY UMR BLOCK POINTER
                                             BEQ     62$                ; BR IF NONE
                                             SWTCHO  #UMRPTR            ; SET TO WRITE FROM HERE
                                             CLR     UMRPTR             ; MAKE A ZERO
                                             PUTAD   R2,#2              ; CLEAR OUT THE POINTER
                                             SWTCHO                     ; RESET TO STANDARD OUTPUT BUFFER
                                             MOV     R0,R2              ; COPY UMR BLOCK ADDRESS FOR DELINKING
                                             CALL    $DEUMR             ; DELINK THE UMR BLOCK
                                             MOV     #M.LGTH,R1         ; LENGTH OF UMR BLOCK
                                             CALL    $DEA16             ; DEALLOCATE THE BLOCK
                                     62$:    MOV     (SP)+,R0           ; RECOVER THE CPU NUMBER
                                             DEC     R0                 ; COUNT DOWN
                                             BGT     60$                ; BR IF MORE CPU'S
                                             BR      70$                ; FINISHED
                                             .ENDC

004430  010200                       65$:    MOV     R2,R0              ; COPY UMR BLOCK ADDRESS
004432  001406                               BEQ     70$                ; IF ZERO, NO UMR'S TO WORRY ABOUT
004434                                       CALL    $DEUMR             ; DE-ASSIGN UMR'S
004440  012701  000014                       MOV     #M.LGTH,R1         ; GET UMR BLOCK LENGTH
004444                                       CALL    $DEA16             ; DE-ALLOCATE UMR BLOCK

004450                               70$:
                                             .IF DF   R$$MPL                                                ;[MP01]

                                             SWTCHO                     ; RESET BUFFER                      ;[MP01]
                                             CLR     $BFR               ; INIT LOCATION                     ;[MP01]
                                             PUTRC   .NTLPT,#2          ; INIT NTL HOME BLOCK POINTER       ;[MP01]
                                             PUTRC   .FILHD,#2          ; INIT FILTER BLOCK POINTER         ;[MP01]
                                             PUTRC   .OBJHD,#2          ; INIT OBJECT BLOCK POINTER         ;[MP01]
                                             PUTRC   .PSIPT,#2          ; INIT PSI HOME BLOCK               ;[MP01]
                                             PUTRC   .CCBLH,#2          ; INIT CCB LISTHEAD                 ;[MP01]
                                             PUTRC   .SDBLH,#2          ; INIT SDB LISTHEAD                 ;[MP01]
                                             MOV     .SDBLH,R1          ; COPY SDB LISTHEAD                 ;[MP01]
                                             ADD     #2,R1              ; POINT TO SDBLH+2                  ;[MP01]
                                             PUTRC   R1,#2              ; INIT SDBLH+2                      ;[MP01]
                                             PUTRC   .RDBLH,#2          ; INIT RDB LISTHEAD                 ;[MP01]
                                             MOV     .RDBLH,R1          ; COPY RDB LISTHEAD                 ;[MP01]
                                             ADD     #2,R1              ; POINT TO RDBLH+2                  ;[MP01]
                                             PUTRC   R1,#2              ; INIT RDBLH+2                      ;[MP01]
                                             .ENDC                                                          ;[MP01]

004450  000241                               CLC                        ; INDICATE SUCCESS

004452                                       RETURN

                                     ;
                                     ; ERROR CONDITION
                                     ;
004454                               101$:   CRASH                      ; SOMEBODY UNLINKED THE NTPOOL PCB!

                                     ;
                                     ; UNLOAD RSX11M DEVICES WHICH ARE ALSO NETWORK PROCESSES
                                     ;
004456  016767  000000G 000000G      XDCB:   MOV     $DEVHD,$BFR        ; GET RSX11M DEVICE POINTER
004464  000241                       10$:    CLC                        ; CLEAR C-BIT
004466  016767  000000G 000000G              MOV     $BFR,$RADDR        ; GET NEXT DEVICE IN LIST
```

```
                                                               N  1
VRGB        CREATED BY  MACRO  ON 30-JUL-85 AT 10:55       PAGE 6
SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL   VALUE              REFERENCES

R$$MPL  = ******             26-962
R$$11D  = ******             20-685
SCEX      000124 R           6-116     #11-317
SCIR      001640 R           6-145     #23-771
SLIN      001544 R           6-125     #22-742
SLTA      000774 R           6-122      6-131      6-143      14-411     18-569     #19-589
SPRO      000670 R           6-120     #17-534
SPSAV     000250 R          #7-182     *9-268      29-1122
SYSFDB  = ******   GX       *9-287     *9-288     *24-869    *24-870    *24-884    *24-885    *26-969    *26-996    *35-1303
                            *35-1313
S$$BAS  = ******             8-209      8-209      8-210      8-210      8-211      8-211      8-212      8-212      8-213
                             8-213      8-214      8-214      8-215      8-215      8-216      8-216      8-217      8-217
                             8-218      8-218      8-219      8-219      8-220      8-220      8-221      8-221      8-222
                             8-222      8-223      8-223      8-224      8-224      8-225      8-225      8-226      8-226
                             8-227      8-227      8-228      8-228      8-229      8-229      8-230      8-230      8-231
                             8-231      8-232      8-232      8-233      8-233      8-234      8-234      8-235      8-235
                             8-236      8-236      8-237      8-237
TSKSCH    002652 R           26-954    #26-994
TY$CHA  = 000400            #5-63
TY$EVE  = 001000            #5-63
TY$STA  = 002000            #5-63
TY$SUM  = 000000            #5-63
T.NAM     000006             26-1004    26-1006
T.RCVL    000012             26-960     35-1304   *35-1306    35-1308   *35-1309
T.TCBL    000030            *26-998     26-1000
UCBBUF    000252 R          #7-186      24-869     24-871
UNIT      000244 R          #7-173     *24-867    *24-879     24-880    *29-1088    29-1097    29-1099   *29-1102    29-1113
US.MNT  = 000100             24-875
U.CW1     000010             24-873
U.STS     000005             24-875
VF.ADD  = ******   GX        14-408
VF.PEM  = ******   GX        14-394
WC.CNT  = 000400            #5-63
WC.EVE  = 004000            #5-63
WC.TRI  = 002000            #5-63
WC.UNT  = 001000            #5-63
ZF.DDM  = ******   GX        19-594
ZF.DLC  = ******   GX        19-629
ZF.KMX  = ******   GX        19-606
ZF.LMC  = ******   GX        19-598
ZF.MUX  = ******   GX        19-602     34-1275
Z.DAT   = ******   GX        17-545
Z.FLG   = ******   GX        15-473     19-594     19-598     19-602     19-606     19-629     34-1261
Z.LLN   = ******   GX        24-828
Z.MAP   = ******   GX        24-831     32-1200
Z.NAM   = ******   GX        28-1053    34-1262
Z.PCB   = ******   GX        14-401     17-541     24-797     24-864
$BFR    = ******   GX        9-287      9-288     *24-857     24-859     24-884     26-960     26-972     26-997     35-1304
                            *35-1306   *35-1309
$BITS     002744 R           17-535     22-743    #27-1021
$CBOMG  = ******   GX        32-1206    34-1274    34-1284
$CMAND  = ******   GX        14-406
```

```
                                                                              N  2
**FILE**ID**VSTB





VV      VV     SSSSSSSS    TTTTTTTTTT   BBBBBBBB
VV      VV     SSSSSSSS    TTTTTTTTTT   BBBBBBBB
VV      VV   SS                TT       BB      BB
VV      VV   SS                TT       BB      BB
VV      VV   SS                TT       BB      BB
VV      VV   SS                TT       BB      BB
VV      VV     SSSSSS          TT       BBBBBBBB
VV      VV     SSSSSS          TT       BBBBBBBB
VV      VV           SS        TT       BB      BB
VV      VV           SS        TT       BB      BB
  VV  VV             SS        TT       BB      BB       ....
  VV  VV             SS        TT       BB      BB       ....
    VV       SSSSSSSS          TT       BBBBBBBB         ....
    VV       SSSSSSSS          TT       BBBBBBBB         ....




LL             SSSSSSSS    TTTTTTTTTT
LL             SSSSSSSS    TTTTTTTTTT
LL           SS                TT
LL           SS                TT
LL           SS                TT
LL           SS                TT
LL             SSSSSS          TT
LL             SSSSSS          TT
LL                   SS        TT
LL                   SS        TT
LL                   SS        TT
LL                   SS        TT
LLLLLLLLLL   SSSSSSSS          TT
LLLLLLLLLL   SSSSSSSS          TT
```

```
   517                                  .SBTTL  SHOW COMMAND
   518                          ;
   519                          ;  SHOW COMMAND
   520                          ;
   521 000000                           STATE$  SHO,&P$SHO
   522 000000                             TRAN$  !SHOID,          SHO1
   523 000000                             TRAN$  !SHOTYP,         SHO2
   524                          ;
   525 000000                           STATE$  SHO1
   526 000000                             TRAN$  !SHOTYP,         SHOOPT
   527 000000                             TRAN$  $LAMDA,          SHOOPT
   528                          ;
   529 000000                           STATE$  SHO2,&P$SHO
   530 000000                             TRAN$  !SHOID,          SHOOPT
   531 000000                             TRAN$  $LAMDA,          DONE
   532                          ;
   533                          ;  SHOW ENTITY IDENTIFICATION
   534                          ;
   535 000000                           STATE$  SHOID
   536 000000                             TRAN$  !IDALI,          $EXIT,     ,          LD.ALI,      RQBST
   537 000000                             TRAN$  !IDALAL,         $EXIT,     ,          LD.ALI,      RQBST
   538 000000                             TRAN$  !IDKNAL,         $EXIT,     ,          LD.ALI,      RQBST
   539 000000                             TRAN$  !IDCIR,          $EXIT,     ,          LD.CIR,      RQBST
   540 000000                             TRAN$  !IDALCI,         $EXIT,     ,          LD.CIR,      RQBST
   541 000000                             TRAN$  !IDKNCI,         $EXIT,     ,          LD.CIR,      RQBST
   542 000000                             TRAN$  !IDLIN,          $EXIT,     $XMISS,    LD.LIN,      RQBST
   543 000000                             TRAN$  !IDKNLI,         $EXIT,     ,          LD.LIN,      RQBST
   544 000000                             TRAN$   IDACLI,         $EXIT,     ,          LD.LIN,      RQBST
   545 000000                             TRAN$  !IDACLO,         $EXIT,     ,          LD.LOG,      RQBST
   546 000000                             TRAN$  !IDKNLO,         $EXIT,     ,          LD.LOG,      RQBST
   547 000000                             TRAN$  !IDLOG,          $EXIT,     ,          LD.LOG,      RQBST
   548 000000                             TRAN$  !IDNOD,          $EXIT,     ,          LD.NOD,      RQBST
   549 000000                             TRAN$  !IDKNNO,         $EXIT,     ,          LD.NOD,      RQBST
   550 000000                             TRAN$  !IDLONO,         $EXIT,     ,          LD.NOD,      RQBST
   551 000000                             TRAN$  !IDKNOB,         $EXIT,     ,          LD.OBJ,      RQBST
   552 000000                             TRAN$  !IDPRO,          $EXIT,     ,          LD.PRO,      RQBST
   553 000000                             TRAN$  !IDACPR,         $EXIT,     ,          LD.PRO,      RQBST
   554 000000                             TRAN$  !IDKNPR,         $EXIT,     ,          LD.PRO,      RQBST
   555 000000                             TRAN$  !IDMOD,          $EXIT,     ,          LD.MOD,      RQBST
   556 000000                             TRAN$  !IDOBJ,          $EXIT,     ,          LD.OBJ,      RQBST
   557 000000                             TRAN$  !IDSYS,          $EXIT,     ,          LD.CEX,      RQBST
   558                          ;
   559                          ;  SHOW INFORMATION TYPE
   560                          ;
   561 000000                           STATE$  SHOTYP
   562 000000                             TRAN$  &K$CHA,          $EXIT,     ,          TY$CHA,      $TYPE
   563 000000                             TRAN$  &K$EVE,          $EXIT,     ,          TY$EVE,      $TYPE
   564 000000                             TRAN$  &K$SUM,          $EXIT,     ,          TY$SUM,      $TYPE
   565 000000                             TRAN$  &K$STUS,         $EXIT,     ,          TY$STA,      $TYPE
   566                          ;
   567                          ;     SHOW OPTIONS
   568                          ;
   569 000000                           STATE$  SHOOPT
   570 000000                             TRAN$  $LAMDA,          SHOALI,    $IFALI
   571 000000                             TRAN$  $LAMDA,          SHOLOG,    $IFLOG
   572 000000                             TRAN$  $LAMDA,          SHOXAC,    $IFXAC
   573 000000                             TRAN$  $LAMDA,          SHOXPR,    $IFXPR
```

```
1184 000000                              STATE$ ,&P$NAD
1185 000000                               TRAN$  $DNUMB,       ,         $ARAD1
1186 000000                              STATE$
1187 000000                               TRAN$ <'.>,          OPA
1188 000000                               TRAN$ $LAMDA,        $EXIT,    $STNAD
1189 000000                              STATE$ OPA
1190 000000                               TRAN$ $DNUMB,        ,         $ARAD2
1191 000000                               STATE$
1192 000000                                TRAN$ $LAMDA,       $EXIT,    $STNAD
1193
1194                              ;
1195                              ; NODE ID
1196                              ;
1197 000000                              STATE$ OPNOD
1198 000000                               TRAN$ &K$NOD
1199 000000                              STATE$ ,&P$NID
1200 000000                               TRAN$ !NODID,        $EXIT,    $STLND
1201                              ;
1202                              ; NODE NAME
1203                              ;
1204 000000                              STATE$ OPNAM
1205 000000                               TRAN$ &K$NAM
1206 000000                              STATE$ ,&P$NAM
1207 000000                               TRAN$ $RAD50,        $EXIT,    $STNAM
1208                              ;
1209                              ; NODE STATE
1210                              ;
1211 000000                              STATE$ OPNOST
1212 000000                               TRAN$ &K$STAE
1213 000000                              STATE$ ,&P$NOST
1214 000000                               TRAN$ &K$OFF,        $EXIT
1215 000000                               TRAN$ &K$ON,         FIX,      ,          VF.ON,         $VFLAG
1216 000000                              STATE$ FIX
1217 000000                               TRAN$ &K$FIX,        $EXIT,    ,          VF.FIX,        $VFLAG
1218 000000                               TRAN$ &K$UNF,        $EXIT
1219 000000                               TRAN$ $LAMDA,        $EXIT,    ,          VF.FIX,        $VFLAG
1220                              ;
1221                              ; OBJECT NAME
1222                              ;
1223 000000                              STATE$ OPOBNA
1224 000000                               TRAN$ &K$NAM
1225 000000                              STATE$ ,&P$OBNA
1226 000000                               TRAN$ $RAD50,        $EXIT,    $ONAM
1227                              ;
1228                              ; OWNER
1229                              ;
1230 000000                              STATE$ OPOWN
1231 000000                               TRAN$ &K$OWN
1232 000000                              STATE$ ,&P$OWN
1233 000000                               TRAN$ $STRNG,        $EXIT,    $OWN
1234                              ;
1235                              ; PARTITION
1236                              ;
1237 000000                              STATE$ OPPAR
1238 000000                               TRAN$ &K$PAR
1239 000000                              STATE$ ,&P$PAR
1240 000000                               TRAN$ $STRNG,        $EXIT,    $STPAR,    LS.OPT,        RQBST
```

```
                                  ;
                                  ; UNCONDITIONALLY REJECT THE TRANSITION
                                  ;
000126                            REJECT:
000126  062716  000002                    ADD     #2,(SP)                 ;REJECT TRANSITION
000132                                    RETURN
```

```
                                                                N  6
VSTB        CREATED BY  MACRO  ON 29-JUN-85 AT 03:03       PAGE 9

MACRO CROSS REFERENCE                                      CREF    04.00

MACRO NAME     REFERENCES

CALL            19-1643
DBGTP$         #6-108       #7-233       #7-234       #7-235       #7-236       #7-237       #8-255       #8-256       #8-258       #8-259
               #8-260       #8-261       #8-263       #8-264       #8-265       #8-266       #8-267       #8-268       #8-269       #8-277
               #8-283       #8-284       #8-291       #8-299       #8-309       #8-310       #8-311       #8-312       #8-313       #8-314
               #8-325       #8-326       #8-327       #8-328       #8-329       #8-335       #8-341       #8-347       #9-361       #9-362
               #9-363       #9-364       #9-366       #9-367       #9-368       #9-369       #9-370       #9-371       #9-372       #9-378
               #9-385       #9-394       #9-395       #9-396       #9-397       #9-398       #9-399       #9-400       #9-401       #9-409
               #9-415       #9-416       #9-417       #9-418       #9-419       #9-420       #9-421       #9-433       #9-434       #9-435
               #9-436       #9-437       #9-438       #9-439       #9-450       #9-451       #9-452       #9-453       #9-454       #9-455
               #9-456       #9-457       #9-458       #9-459       #9-460       #9-461       #9-462       #9-473       #9-482       #9-488
               #9-489       #9-490       #9-496       #9-500       #9-506       #9-507       #9-508       #9-514       #9-515       #10-523
               #10-527      #10-531      #10-537      #10-538      #10-539      #10-540      #10-541      #10-542      #10-543      #10-544
               #10-545      #10-546      #10-547      #10-548      #10-549      #10-550      #10-551      #10-552      #10-553      #10-554
               #10-555      #10-556      #10-557      #10-563      #10-564      #10-565      #10-571      #10-572      #10-573      #10-574
               #10-575      #10-576      #10-582      #10-583      #10-589      #10-590      #10-596      #10-597      #10-603      #10-604
               #10-605      #10-611      #10-612      #11-671      #11-754      #12-794      #12-806      #12-809      #12-810      #12-813
               #12-816      #12-819      #12-820      #12-831      #12-836      #12-841      #12-846      #12-849      #12-859      #12-862
               #12-865      #12-871      #12-872      #12-878      #12-884      #12-885      #12-889      #12-892      #12-900      #12-904
               #12-905      #12-918      #12-921      #12-922      #12-936      #12-939      #13-970      #13-989      #13-1011     #13-1012
               #13-1013     #13-1017     #13-1028     #13-1038     #13-1051     #13-1057     #13-1067     #13-1088     #13-1109     #13-1122
               #13-1123     #13-1126     #13-1134     #13-1142     #13-1188     #13-1215     #13-1218     #13-1219     #13-1260     #13-1265
               #13-1274     #13-1284     #13-1304     #13-1307     #13-1310     #13-1385     #13-1388     #13-1391     #13-1415     #13-1427
               #13-1442     #13-1443     #13-1451     #13-1454     #14-1488     #14-1496     #14-1510     #14-1515     #14-1516     #14-1526
               #15-1558     #15-1559     #15-1560     #15-1566     #15-1581     #15-1582
ISTAT$         #5-61         6-108
KEYWD$         #5-61         6-112       #6-113       #6-114       #6-115       #6-116       #6-117       #6-118       #6-119       #6-120
               #6-121       #6-122       #6-123       #6-124       #6-125       #6-126       #6-127       #6-128       #6-129       #6-130
               #6-131       #6-132       #6-133       #6-134       #6-135       #6-136       #6-137       #6-138       #6-139       #6-140
               #6-141       #6-142       #6-143       #6-144       #6-145       #6-146       #6-147       #6-148       #6-149       #6-150
               #6-151       #6-152       #6-153       #6-154       #6-155       #6-156       #6-157       #6-158       #6-159       #6-160
               #6-161       #6-162       #6-163       #6-164       #6-165       #6-166       #6-167       #6-168       #6-169       #6-170
               #6-171       #6-172       #6-173       #6-174       #6-175       #6-176       #6-177       #6-178       #6-179       #6-180
               #6-181       #6-182       #6-183       #6-184       #6-185       #6-186       #6-187       #6-188       #6-189       #6-190
               #6-191       #6-192       #6-193       #6-194       #6-195       #6-196       #6-197       #6-198       #6-199       #6-200
               #6-201       #6-202       #6-203       #6-204       #6-205       #6-206       #6-207       #6-208       #6-209       #6-210
               #6-211       #6-212       #6-213       #6-214       #6-215       #6-216       #6-217       #6-218       #6-219       #6-220
MTRAN$         #6-108
PRMPT$         #5-61
RETURN          17-1598      17-1605      18-1621      18-1625      19-1644      20-1666      21-1673      22-1686
STATE$         #5-61         7-231       #7-241        8-253       #8-275       #8-281       #8-289       #8-297       #8-304       #8-307
               #8-318       #8-323       #8-333       #8-339       #8-345        9-358       #9-376       #9-380       #9-383       #9-390
               #9-392       #9-407       #9-413       #9-426       #9-431       #9-443       #9-448       #9-466       #9-471       #9-477
               #9-480       #9-486       #9-494       #9-498       #9-504       #9-512        10-521      #10-525      #10-529      #10-535
               #10-561      #10-569      #10-580      #10-587      #10-594      #10-601      #10-609       11-625      #11-627      #11-632
               #11-634      #11-639      #11-641      #11-646      #11-648      #11-653      #11-655      #11-660      #11-662      #11-667
               #11-669      #11-675      #11-677      #11-682      #11-684      #11-689      #11-691      #11-696      #11-698      #11-703
               #11-705      #11-710      #11-712      #11-717      #11-719      #11-724      #11-726      #11-731      #11-733      #11-738
               #11-740      #11-745      #11-747      #11-752      #11-755      #11-760      #11-762      #11-767      #11-769      #11-774
                12-788      #12-790      #12-792      #12-795      #12-800      #12-802      #12-804      #12-807      #12-811      #12-814
               #12-817      #12-824      #12-829      #12-832      #12-834      #12-837      #12-839      #12-842      #12-844      #12-847
               #12-853      #12-855      #12-857      #12-860      #12-863      #12-869      #12-876      #12-880      #12-882      #12-887
               #12-890      #12-896      #12-898      #12-902      #12-907      #12-914      #12-916      #12-919      #12-923      #12-925
```

```
VSYM      CREATED BY  MACRO  ON 29-JUN-85 AT 03:07      PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE               REFERENCES

AB       = 000000   G        #5-158
BUFUMP   = 172354   G        #5-63
CMODE    = 140000   G        #5-63
I$$AS    = ******             5-63        5-64
KISAR0   = 172340   G        #5-63
KISAR6   = 172354   G        #5-63
LB.ADR     000006   G        #5-166
LB.APR   = 000013   G        #5-169
LB.SIZ     000010   G        #5-167
LB.USE     000012   G        #5-168       5-169
LF.ACT   = 100000   G        #5-65
LF.BRO   = 000400   G        #5-65
LF.BWT   = 000007   G        #5-65
LF.ENA   = 002000   G        #5-65
LF.LPB   = 001000   G        #5-65
LF.MDC   = 000100   G        #5-65
LF.MFL   = 004000   G        #5-65
LF.MTP   = 000020   G        #5-65
LF.PAC   = 000200   G        #5-65
LF.RDY   = 040000   G        #5-65
LF.REA   = 010000   G        #5-65
LF.SER   = 000040   G        #5-65
LF.TIM   = 000010   G        #5-65
LF.UNL   = 020000   G        #5-65
LF.X2P   = 000000   G        #5-65
LN.CLO   = 000000   G        #5-65
LN.DUM   = 000005   G        #5-65
LN.LOA   = 000004   G        #5-65
LN.LOO   = 000003   G        #5-65
LN.OAU   = 000003   G        #5-65
LN.OFF   = 000001   G        #5-65
LN.ON    = 000000   G        #5-65
LN.OOP   = 000004   G        #5-65
LN.OPE   = 000001   G        #5-65
LN.REF   = 000002   G        #5-65
LN.SER   = 000002   G        #5-65
LN.STA   = 000017   G        #5-65
LN.SUB   = 000360   G        #5-65
LN.TRI   = 000006   G        #5-65
L.COST     000015   G        #5-65
L.CTL      000012   G        #5-65
L.CVA      177776   G        #5-65
L.DDM      000002   G        #5-65
L.DDS      000004   G        #5-65
L.DLC      000003   G        #5-65
L.DLM      000006   G        #5-65
L.DLS      000010   G        #5-65
L.FLG      000000   G        #5-65
L.KRBA     000016   G        #5-65
L.LEN    = 000022   G        #5-65
L.MPF      000022   G        #5-65
L.NMST     000020   G        #5-65
```

```
000200                                  CALL    $CLDEQ          ; CLOSE TEMPLATE FILE
000204  012667  000002'         30$:    MOV     (SP)+,$FLAG2    ; RESTORE $FLAG2
000210  162767  000002  000036'         SUB     #2,$SZPTR       ; RESET SIZE TABLE POINTER
000216                                  RETURN

                                ;
                                ; PROCESS TEMPLATE FILE FROM "TMPAC2" OVERLAY
                                ;
000220                          $TMFIX::
000220                                  CALL    $TMFIL          ; PROCESS SECONDARY TEMPLATE
000224                                  CALLR   $FILEX          ; RETURN TO "$FILE" SUBROUTINE

                                ;
                                ; ACCESS BINARY FILE AND RELOAD TPARS TABLES
                                ;
000230                          .ABFI::
000230                                  CALL    $ABFI           ; PROCESS FILE NAME OPERAND
000234                                  CALLR   $TPARS          ; RELOAD TPARS TABLES
```

```
VTMP - VNP TEMPLATE PROCESSING  MACRO V05.03b Saturday 29-Jun-85 03:08  Page 18-2
Symbol table

SY:VTMP.V2,[132,134]VTMP/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VTMP
```

```
                                                .SBTTL  FNDADD - FIND LINE TABLE
                                        ;+
                                        ; FNDADD - FIND LINE TABLE ADDRESS
                                        ;
                                        ; INPUTS:
                                                R1 = VECTOR ADDRESS
                                        ;
                                        ; OUTPUTS:
                                                $ADDR = LINE TABLE ADDRESS
                                                R1 = PRESERVED
                                                R2 = DESTROYED
                                        ;
                                        ;-

000320                                  FNDADD: SAVRG   <R1>                    ; PRESERVE R1
000322                                          GETRV   R1,#4                   ; READ IN VECTOR
000340  016702  000000G                         MOV     $BFR,R2                 ; POINT AT ICB OR LINE TABLE
000344  010267  000000G                         MOV     R2,$ADDR                ; ASSUME VECTOR POINTS AT LINE TABLE
000350  016701  000000G                         MOV     $ICBLN,R1               ; GET ICB SIZE
000354                                          GETRV   R2,R1                   ; READ IN PART OF POSSIBLE ICB
000370  012702  000000G                         MOV     #$BFR,R2                ; POINT TO ICB
000374  005767  000000G                         TST     .MGMGE                  ; MAPPED SYSTEM?
000400  100404                                  BMI     3$                      ; BR IF NO

000402  066702  000000G                         ADD     $ICBLN,R2
000406  162702  000004                          SUB     #4,R2                   ; POINT TO KEY WORD IN ICB

000412  022712  004537                  3$:     CMP     #4537,(R2)              ; IF "JSR R5,X", THEN LINE TABLE ADDRESS
000416  001405                                  BEQ     10$                     ; ...
000420  022722  000137                          CMP     #137,(R2)+              ; IF "JMP LINE-TABLE", THEN LINE TABLE
000424  001006                                  BNE     101$                    ; ..ADDRESS IS NEXT WORD, ELSE ERROR
000426  011267  000000G                 5$:     MOV     (R2),$ADDR              ; SAVE REAL LINE TABLE ADDRESS
000432                                  10$:    RESRG   <R1>                    ; RESTORE R1
000434  005702                                  TST     R2                      ; SUCCESS ?
000436  001401                                  BEQ     101$                    ; IF EQ, NO

000440                                          RETURN
                                        ;
                                        ; ERROR CONDITIONS
                                        ;
000442  000004                          101$:   IOT                             ; WHAT IS R2 POINTING AT ????

        000001                                  .END
```

```
                                      ;
                                      ; PUT 22BIT VIRTUAL ADDRESS
                                      ;
 001326                               PUTAD:  GETNUM  <BIAS>,$RBLCK,X104
 001354                                       GETNUM  <VIRTUAL ADDRESS>,$RADDR,X104
 001402                                       GETNUM  <NUMBER OF BYTES>,$RSIZE,X104
 001430                                       PUTAD
 001434                               X104:   RETURN
                                      ;
                                      ; PUT 16BIT VIRTUAL ADDRESS
                                      ;
 001436                               PUTRC:  GETNUM  <VIRTUAL ADDRESS>,$RADDR,X105
 001464                                       GETNUM  <NUMBER OF BYTES>,$RSIZE,X105
 001512                                       PUTRC
 001516                               X105:   RETURN
                                      ;
                                      ; DUMP BYTES FROM BUFFER
                                      ;
                                              .ENABL  LSB
 001520  016703  000000G              DUMPB:  MOV     $RSIZE,R3
 001524                                       TYPEN   R3,LF
 001562                                       TYPES   < BYTES IN BUFFER>,CR
 001612                                       TYPES   < >,LFCR
 001642  005703                               TST     R3                       ; ANYTHING IN BUFFER ?
 001644  001461                               BEQ     50$                      ; IF EQ, NO
 001646  012705  000000G                      MOV     #$BFR,R5                 ; POINT AT BUFFER
 001652  012704  000020               10$:    MOV     #16.,R4                  ; 16. BYTES TO A LINE
 001656                               20$:    TYPEN   (R5)+,NO,OB              ; OUTPUT OCTAL BYTE
 001716                                       TYPES   < >,NO                   ; PUT IN SPACE
 001746  005303                               DEC     R3                       ; MORE IN BUFFER ?
 001750  001417                               BEQ     50$                      ; IF EQ, NO
 001752  005304                               DEC     R4                       ; MORE ROOM ON LINE ?
 001754  003340                               BGT     20$                      ; IF GT, YES
 001756                                       TYPEB   #NEWLIN,NO,#2            ; ELSE, TYPE NEW LINE
 002006  000721                               BR      10$                      ; AND CONTINUE
 002010                               50$:    TYPEB   #NUL,CR,#1               ; INSERT CARRIAGE RETURN
 002040                                       RETURN
                                              .DSABL  LSB

 002042     012     015               NEWLIN: .BYTE   12,15
 002044     040     011               TAB:    .BYTE   40,11
 002046     040     054     073       DELIM:  .ASCIZ  " ,;"
 002051     000
 002052     000                       NUL:    .BYTE   0
                                              .EVEN
                                      ;
                                      ; DUMP WORDS
                                      ;
                                              .ENABL  LSB
 002054  016703  000000G              DUMPW:  MOV     $RSIZE,R3                ; GET SIZE OF BUFFER IN BYTES
 002060  062703  000001                       ADD     #1,R3                    ; ROUND TO NEXT WORD
 002064  042703  000001                       BIC     #1,R3                    ; ...
 002070  006203                               ASR     R3                       ; CONVERT TO WORDS
 002072                                       TYPEN   R3,LF                    ; GIVE LENGTH OF BUFFER
 002130                                       TYPES   < WORDS IN BUFFER>,CR    ; ...
 002160                                       TYPES   < >,LFCR                 ; ...
 002210  005703                               TST     R3                       ; ANYTHING IN BUFFER ?
```

```
    141 000170                          RETURN
    142
```

```
                                                                                  N 13
VUNB - VNP UNLOAD BLOCK SCANNIN MACRO V05.03b Saturday 29-Jun-85 03:09  Page 7-1
Symbol table

A$$CHK= 000000            F.NRBD= ****** GX         L$$P11= 000001            P$$P45= 000000            $ALN18= ****** GX
A$$CPS= 000000            F.RSIZ= ****** GX         L$$11R= 000000            P$$WRD= 000000            $BFR  = ****** GX
A$$PRI= 000000            F.URBD= ****** GX         M$$CRB= 000124            Q$$OPT= 000010            $EROUT= ****** GX
A$$TRP= 000000            G$$TPP= 000000            M$$CRX= 000000            R$$DER= 000000            $GETAD= ****** GX
C$$CKP= 000000            G$$TSS= 000000            M$$FCS= 000000            R$$K11= 000001            $PUTAD= ****** GX
C$$ORE= 000400            G$$TTK= 000000            M$$MGE= 000000            R$$SND= 000000            $RSIZE= ****** GX
C$$RSH= 177564            G$$WRD= 000000            M$$NET= 000000            R$$11M= 000000            $SAVRG= ****** GX
D$$BUG= 177514            I$$RAR= 000000            M$$OVR= 000000            SYSFDB= ****** GX         $SAVVR= ****** GX
D$$ISK= 000000            I$$RDN= 000000            N$$ACC= 000001            S$$WRG= 000000            $UNBAD= ****** GX
D$$L11= 000001            K$$CNT= 177546            N$$BUF= 000001            S$$YSZ= 007600            $UNBLK= ****** GX
D$$YNC= 000000            K$$CSR= 177546            N$$LDV= 000001            T$$KMG= 000000            $UNBSZ= ****** GX
D$$YNM= 000000            K$$LDC= 000000            N$$MCP= 000001            T$$MIN= 000000            $UNEND= ****** GX
ER1     000446R           K$$TPS= 000074            N$$MLL= 000001            V$$CTR= 001000            $VUNB   000000RG
ER2     000524R           LD$LP = 000000            N$$MOV= 000010            X$$DBT= 000000            $$    = 177777
E$$XPR= 000000            LINK    000166R           N$$NCT= 000001            $ADRX = ****** GX         $$$   = 000450R
FATAL = ****** GX         L$$ASG= 000000            N$$PEM= 000001            $ADRZ = ****** GX         $$$$  = 000524R
F$$LVL= 000001            L$$DRV= 000000

. ABS.  000000     000    (RW,I,GBL,ABS,OVR)
        000526     001    (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 9928 Words  ( 39 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:09.25
SY:VUNB.V2,[132,134]VUNB/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VUNB
```

```
                                        .SBTTL  GETIND - GET PDV INDEX
                                ;+
                                ; GETIND - GET PDV INDEX FROM PDV ADDRESS
                                ;
                                ; INPUTS:
                                ;       R4 = PDV ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       R4 = PDV INDEX
                                ;       ALL OTHER REGISTERS PRESERVED
                                ;
                                ;-

000522                          GETIND:
000522  010546                          MOV     R5,-(SP)        ; FREE UP A REGISTER
000524  016746  000000G                 MOV     $PDVNM,-(SP)    ; GET # OF PDV ENTRIES
000530  016705  000000G                 MOV     $PDVTA,R5       ; POINT TO PDV MAPPING TABLE
000534  022504                  10$:    CMP     (R5)+,R4        ; POINTING AT OUR ENTRY ?
000536  001403                          BEQ     20$             ; IF EQ, YES
000540  005316                          DEC     (SP)            ; ANY MORE ?
000542  003374                          BGT     10$             ; IF GT, YES
000544  000004                          IOT                     ; ELSE, CAN'T FIND PDV INDEX
000546  005726                  20$:    TST     (SP)+           ; CLEAN UP STACK
000550  166705  000000G                 SUB     $PDVTA,R5       ; CALCULATE INDEX
000554  005745                          TST     -(R5)           ; ...
000556  010504                          MOV     R5,R4           ; RETURN INDEX IN R4
000560  012605                          MOV     (SP)+,R5        ; RESOTRE R5
000562                                  RETURN
```

```

;+
;
; DEICP - DEALLCOATE ICB'S (MULTIPROCESSOR ONLY)
;
;
; INPUTS:
;       $SLTA  - SLT ADDRESS
;       .K6TAB - BIAS OF CPU PARTITION TABLE
;
; OUTPUTS:
;       ICB'S DEALLOCATED
;
;-

        .IF DF  R$$MPL                                                  ;[MP01]

DEICP:  CALL    $SAVAL          ; SAVE REG'S                            ;[MP01]
        MOV     $SLTA,R1        ; GET SLT ADDRESS                       ;[MP01]
        MOV     L.KRBA(R1),R1   ; GET KRB ADDRESS                       ;[MP01]
        ADD     #K.PRM,R1       ; ADJUST R1 TO OFFSET 0                 ; ''

        GETRV   R1,#$KRBLN      ; READ KRB
        MOV     #$BFR,R0        ; SET UP BUFFER ADDRESS
        SUB     #K.PRM,R0       ; POINT TO OFFSET 0
        MOV     K.PRM(R0),R0    ; GET ICB ADDRESS/2

10$:    MOV     R0,R2           ; COPY ICB ADDRESS/2
        BIC     #7777.R2        ; FROM CPU NUMBER
        BIC     R2,R0           ; ...
        SWAB    R2              ;
        ASR     R2              ; ...
        ASR     R2              ; ...
        ASR     R2              ; ...
        ASR     R2              ; ...
        MOV     R2,CPU          ; STORE VALUE

        MOV     #$ICAVL,R3      ; GET ICB AVAILABLE LIST
        MOV     CPU,R1          ; GET CPU NUMBER
        ASL     R1              ; MULTIPLY BY 8.
        ASL     R1              ; ...
        ASL     R1              ; ...
        ADD     R1,R3           ; CALCULATE ICB LISTHEAD

        MOV     $ICBLN,R1       ; GET ICB LENGTH
        MOV     CPU,R2          ; COPY CPU NUMBER
        ASL     R2              ; CONVERT TO WORD INDEX
        ADD     .K6TAB,R2       ; GET CPU LOWCR BIAS
        SWTCHI  #ICBVL          ; SET INPUT BUFFER
        GETRV   R2,#2           ; READ CPU BIAS

        MOV     ICBVL,$RBLCK    ; SET UP BIAS
        ASL     R0              ; CREATE ICB VIRTUAL ADDRESS
        SWTCHI  #NXICB          ; SWITCH INPUT BUFFER
        GETAD   R0,#2           ; GET NEXT ICB ADDRESS
        SUB     #2,R0           ; INCLUDE LENGTH FILED
        SWTCHI                  ; RESET BUFFER
        SWTCHO                  ; RESET BUFFER
```

B	1	
C	1	
D	1	
E	1	
F	1	
G	1	
H	1	
I	1	
J	1	
K	1	
L	1	
M	1	
N	1	
B	2	
C	2	
D	2	
E	2	
F	2	VSET
G	2	
H	2	
I	2	
J	2	
K	2	
L	2	
M	2	
N	2	VSTB
B	3	
C	3	
D	3	
E	3	
F	3	
G	3	
H	3	
I	3	
J	3	
K	3	
L	3	
M	3	
N	3	
B	4	
C	4	
D	4	
E	4	
F	4	
G	4	
H	4	
I	4	
J	4	
K	4	
L	4	
M	4	
N	4	
B	5	
C	5	
D	5	
E	5	
F	5	
G	5	
H	5	
I	5	
J	5	
K	5	
L	5	
M	5	
N	5	
B	6	
C	6	
D	6	
E	6	
F	6	
G	6	
H	6	
I	6	
J	6	
K	6	
L	6	
M	6	
N	6	
B	7	
C	7	
D	7	VSYM
E	7	
F	7	
G	7	
H	7	
I	7	
J	7	
K	7	
L	7	
M	7	
N	7	
B	8	
C	8	
D	8	
E	8	
F	8	VTMP
G	8	
H	8	
I	8	
J	8	
K	8	
L	8	
M	8	
N	8	
B	9	
C	9	
D	9	
E	9	
F	9	
G	9	
H	9	
I	9	
J	9	
K	9	
L	9	
M	9	
N	9	
B	10	
C	10	
D	10	
E	10	
F	10	VTMPIN
G	10	
H	10	
I	10	
J	10	
K	10	
L	10	
M	10	
N	10	
B	11	
C	11	
D	11	
E	11	
F	11	VTST
G	11	
H	11	
I	11	
J	11	
K	11	
L	11	
M	11	
N	11	
B	12	
C	12	
D	12	
E	12	
F	12	
G	12	
H	12	
I	12	
J	12	VUMR
K	12	
L	12	
M	12	
N	12	
B	13	
C	13	
D	13	
E	13	
F	13	
G	13	
H	13	
I	13	VUNB
J	13	
K	13	
L	13	
M	13	
N	13	
B	14	
C	14	
D	14	VUNL
E	14	
F	14	
G	14	
H	14	
I	14	
J	14	
K	14	
L	14	
M	14	
N	14	
B	15	
C	15	
D	15	
E	15	
F	15	
G	15	
H	15	
I	15	
J	15	
K	15	
L	15	
M	15	
N	15	
B	16	
C	16	
D	16	
E	16	
F	16	
G	16	
H	16	
I	16	
J	16	
K	16	
L	16	
M	16	

```
   1468 004474   001437                              BEQ     20$                 ; IF ZERO, ALL DONE
   1469 004476                                       GETRV   ,#D.PCB+2           ; READ IN DCB IMAGE
   1470
   1471 004510   022767  054523  000000C             CMP     #"SY,D.NAM+$BFR     ; SKIP SHORT DCB'S
   1472 004516   001762                              BEQ     10$                 ; IF SY: - BRANCH
   1473 004520   022767  041114  000000C             CMP     #"LB,D.NAM+$BFR     ; LB: DCB ?
   1474 004526   001756                              BEQ     10$                 ; IF YES - BRANCH
   1475 004530   022767  046103  000000C             CMP     #"CL,D.NAM+$BFR     ; CL: DCB ?
   1476 004536   001752                              BEQ     10$                 ; IF YES BRANCH
   1477 004540   022767  044524  000000C             CMP     #"TI,D.NAM+$BFR     ; TI: DCB ?
   1478 004546   001746                              BEQ     10$                 ; IF YES - BRANCH
   1479
   1480 004550   026764  000000C 000000G             CMP     D.PCB+$BFR,Z.PCB(R4) ; SAME PCB AS OUR PROCESS?
   1481 004556   001342                              BNE     10$                 ; IF NE, NO
   1482 004560   005067  000000C                     CLR     D.DSP+$BFR          ; MARK DEVICE AS ''UNLOADED''
   1483 004564   005067  000000C                     CLR     D.PCB+$BFR          ;       ...
   1484 004570                                       PUTRC                       ; WRITE PCB BACK TO SYS IMG
   1485 004574                             20$:      RETURN                      ; BACK TO USER MODE AND THEN TO CALLER
   1486
   1487                                  ;
   1488                                  ; DE-ALLOCATE OBJECT TABLES
   1489                                  ;
   1490 004576                             XOBJTB:
   1491 004576   016700  000000G                     MOV     $OBJHD,R0           ; POINT AT START OF OBJECT TABLE LIST
   1492 004602   001425                              BEQ     30$                 ; IF EQ, END OF LIST
   1493
   1494 004604                             10$:      CEACC$  R0,R3               ; CONVERT TO MAPPED ADDRESS
   1495
   1496 004614                             20$:      GETAD   R3,#4               ; GET THE OBJECT BLOCK
   1497 004632   012701  000012                      MOV     #O.LEN,R1           ; GET LENGTH OF OBJECT BLOCK
   1498 004636   016746  000000G                     MOV     $BFR,-(SP)          ; SAVE ADDRESS OF NEXT BLOCK
   1499 004642                                       CALL    $DEPOL              ; DE-ALLOCATE THE OBJECT BLOCK
   1500 004646   012600                              MOV     (SP)+,R0            ; GET UNMAPPED ADDRESS OF NEXT BLOCK
   1501 004650   001355                              BNE     10$                 ; IF NE, MORE
   1502 004652   005067  000000G                     CLR     $RBLCK              ; ELSE, CLEAN UP
   1503 004656                             30$:      RETURN
   1504
   1505                                  ;
   1506                                  ; DE-ALLOCATE ALIAS TABLES
   1507                                  ;
   1508 004660                             XALITB:   SWTCHI  #$HMBUF             ; USE TEMPORARY BUFFER
   1509 004666                                       GETRV   $DECPT,#D$END       ; READ DECNET HOME BLOCK
   1510 004706                                       SWTCHI                      ; RESET DEFAULT BUFFER
   1511
   1512 004714   016700  000000G                     MOV     $HMBUF+D$ANN,R0     ; POINT AT START OF OBJECT TABLE LIST
   1513 004720   001437                              BEQ     30$                 ; IF EQ, END OF LIST
   1514
   1515 004722                             10$:      CEACC$  R0,R3               ; CONVERT TO MAPPED ADDRESS
   1516
   1517 004732                             20$:      GETAD   R3,#4               ; GET THE ALIAS BLOCK
   1518 004750   016746  000000G                     MOV     $BFR,-(SP)          ; SAVE THE ADDRESS OF NEXT ALIAS
   1519 004754   012701  000000G                     MOV     #$BFR,R1            ; CALCULATE LENGTH OF THIS ALIAS
   1520 004760   062701  000020                      ADD     #A.ACC,R1           ;       POINT AT START OF ACC. INFO.
   1521 004764   112102                              MOVB    (R1)+,R2            ;       LENGTH OF FIRST FIELD
   1522 004766   060201                              ADD     R2,R1               ;       ...
   1523 004770   112102                              MOVB    (R1)+,R2            ;       LENGTH OF SECOND FIELD
   1524 004772   060201                              ADD     R2,R1               ;       ...
```

```
                                                              B  2
VUNL        CREATED BY  MACRO   ON 16-SEP-85 AT 07:31      PAGE 2
SYMBOL  CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE                REFERENCES

FF.CON   = 000001             #5-80
FF.FIL   = 000002             #5-80
FF.HST   = 040000             #5-80
FF.LIN   = 000010             #5-80      5-80
FF.MOD   = 000100             #5-80      5-80
FF.MON   = 000004             #5-80
FF.MSK   = 000077             #5-80      21-1570
FF.PRT   = 000200             #5-80      5-80
FF.QUL   = 000370             #5-80
FF.REM   = 100000             #5-80
FLT        005242 R           #22-1593
FLT16      005242 R            14-739    #22-1592   23-1654
FL.PX1   = ******    GX        8-264      12-571    12-580
FL.PX2   = ******    GX        12-585
FNDADD     002174 R            14-737    #15-825
F.ADD      000016             #5-80
F.CEV      000030             #5-80
F.CIR      000024             #5-80
F.CLS      000002             #5-80      21-1569
F.EVT      000004             #5-80
F.FLG      000014             #5-80
F.LEN      000016             #5-80
F.LIN      000020             #5-80
F.LNK      000000             #5-80
F.MOD      000022             #5-80
F.NRBD   = ******    GX       *17-1047   *20-1225   *20-1227   *23-1773   *23-1778
F.REM      000026             #5-80
F.RSIZ   = ******    GX       *7-208     *8-322     *8-334     *10-442    *10-453    *10-466    *10-479    *10-485    *10-497
                              *11-524    *14-700    *15-827    *15-836    *17-962    *17-1007   *17-1050   *20-1226   *21-1264
                              *21-1271   *21-1394   *21-1469   *21-1496   *21-1509   *21-1517   *21-1538   *21-1547   *21-1567
                              *23-1776   *24-1814   *25-1849
F.SEV      000004             #5-80
F.URBD   = ******    GX       *18-1089   *18-1165   *21-1508   *21-1510   *21-1537   *21-1539   *22-1596   *22-1641
GETIND     000522 R            8-382     #9-410
GETPDV     002150 R            14-654     14-743     14-765    #14-793
ICBLK      000150 R           #6-177      18-1089    18-1118
ICBSAV     000174 R           #6-178     *17-1014   *17-1025    23-1669
ICBSC      003132 R            17-1020    17-1043   #18-1089
ICBSZ      000176 R           #6-179     *17-1015   *18-1161    23-1668
I$$AS    = ******              10-449     10-463     10-476     10-493     11-531     17-1031
KMCB       000114 R           #6-148
K.VCT      000001              14-726     17-936
LF.ACT   = ******    GX        14-651
LF.ENA   = ******    GX        14-651
LF.RDY   = ******    GX        14-651     15-863     21-1290
LF.UNL   = ******    GX        14-778
L.CTL    = ******    GX        15-871     15-871     19-1179    19-1179    20-1220    20-1220
L.DDM    = ******    GX        14-653     14-738     14-760     15-868     15-868     19-1177    19-1177    19-1183    20-1218
                               20-1218
L.DDS    = ******    GX       *19-1181
L.DLC    = ******    GX        12-575     14-760     14-764     14-768     15-865     15-865     19-1183
L.DLM    = ******    GX       *12-578    *14-771
```

```
                                                              C  2
VUNL        CREATED BY  MACRO   ON 16-SEP-85 AT 07:31      PAGE 3
SYMBOL  CROSS REFERENCE                                    CREF    04.00
```

```
     98                                     .SBTTL  LOCAL DATA
     99 000000                              .PSECT  DATA,D                                                           ;[TD001]
    100                                                                                                              ;[TD001]
    101 000000                     LSTHBF:  .BLKW   2                         ; BUFFER FOR LISTHEADS FROM HOME BLOCK ;[TD001]
    102                                                                                                              ;[TD001]
    103 000000                              .PSECT                                                                   ;[TD001]
```

```
Symbol table

A$$CHK= 000000      G$LNK   000000      LP$UP = 010000          M$$CRB= 000124      SF$INC= 000002
A$$CPS= 000000      G$NAM   000002      LP$WTD= 020000          M$$CRX= 000000      SYSFDB= ****** GX
A$$PRI= 000000      G$NML   000015      LP$WTS= 040000          M$$FCS= 000000      S$ADSZ  000007
A$$TRP= 000000      G$$TPP= 000000      LSTHBF  000000R     002 M$$MGE= 000000      S$CTL   000002
CF$BLK= 000102      G$$TSS= 000000      LS$GDN= 000004          M$$NET= 000000      S$DTE   000020
C$DTE   000012      G$$TTK= 000000      LS$OFF= 000000          M$$OVR= 000000      S$FLG   000010
C$FLG   000015      G$$WRD= 000000      LS$STR= 000002          N$$ACC= 000001      S$LEN   000032
C$GNAM  000016      HF$DLM= 000002      LS$UP = 000003          N$$BUF= 000001      S$LFT   000012
C$LCN   000010      HF$GWY= 000010      LS$WT = 000001          N$$LDV= 000001      S$LNK   000000
C$LEN   000016      HF$HOS= 000004      L$ACHN  000012          N$$MCP= 000001      S$OWNR  000030
C$LNK   000000      HF$XDF= 000020      L$APVC  000014          N$$MLL= 000001      S$PKSZ  000004
C$NAM   000002      H$CUG   000010      L$ASVC  000010          N$$MOV= 000010      S$RCL   000013
C$NML   000002      H$DST   000012      L$AUC   000042          N$$NCT= 000001      S$STA   000003
C$PORT  000014      H$D29   000014      L$CHLS  000034          N$$PEM= 000001      S$TIM   000014
C$$CKP= 000000      H$FLG   000000      L$CHTB  000030          PF.PSV= ****** GX   S$TMR   000016
C$$ORE= 000400      H$GLEN  000104      L$CLEN= 000044          PF.XDF= ****** GX   S$WND   000006
C$$RSH= 177564      H$GLT   000044      L$CTEN  000032          PF.XGA= ****** GX   S$$WRG= 000000
DEACUG  000374R     H$GNAM  000050      L$CTIM  000040          P$$P45= 000000      S$$YSZ= 007600
DEADST  000552R     H$GNML= 000020      L$DTEA  000020          P$$WRD= 000000      TC$LNG= 000002
DEALST  002046R     H$GPT   000046      L$DTEL  000017          Q$ACTC  000004      TC$TDA= 000001
DEAPVC  000216R     H$HITS  000034      L$GLEN  000134          Q$AUC   000006      TR$DCE= 000010
D$ACL   000027      H$HLEN  000044      L$LEN   000122          Q$CLEN= 000006      TR$FIL= 000001
D$CUGL  000023      H$LBDA  000070      L$LLCH  000016          Q$CTIM  000000      TR$OVR= 000004
D$CUGN  000022      H$LBDN  000072      L$LNK   000000          Q$ICRE  000014      TR$TRU= 000020
D$DATL  000030      H$LDTE  000002      L$MCHN  000036          Q$LEN   000016      TR$TX = 000002
D$DSL   000022      H$LEN   000042      L$NETW  000114          Q$MXAC  000012      TS.BLK= 171700
D$DST   000010      H$LOTS  000032      L$NIRE  000072          Q$MXVC  000002      TS.CKP= 000200
D$DTEL  000031      H$NETW  000024      L$NLRE  000104          Q$TCLZ  000010      TS.CKR= 000100
D$FLEN  000032      H$NML = 000006      L$NMAC  000076          Q$$OPT= 000010      TS.EXE= 100000
D$GLEN  000032      H$NPT   000022      L$NMAS  000074          R$ACC   000064      TS.HLD= 002000
D$GVBL  000032      H$PTB   000020      L$NMST  000003          R$ACL   000063      TS.MSG= 020000
D$LNK   000000      H$PVC   000006      L$NNRE  000106          R$DAL   000013      TS.NRP= 010000
D$NAM   000004      H$RDTE  000004      L$NRBY  000046          R$DTE   000014      TS.OUT= 000400
D$NOD   000010      H$RNW   000042      L$NRCA  000066          R$LEN   000024      TS.RDN= 040000
D$OBJ   000003      H$SVC   000036      L$NRFS  000100          R$LNK   000000      TS.RUN= 004000
D$PAL   000026      H$TRB   000016      L$NRPK  000056          R$NAM   000002      TS.STP= 001000
D$PRI   000002      H$XAVL  000100      L$NRRE  000105          R$NML   000002      TY$LAB= 000002
D$PRL   000024      H$XBIA  000074      L$NRST  000107          R$NOD   000023      TY$LAP= 000000
D$SHI   000020      H$X29C  000040      L$NTBY  000052          R$NWK   000003      T$CBUF  000004
D$SLO   000016      I$$RAR= 000000      L$NTCA  000070          R$NWL   000104      T$CEND  000006
D$USL   000025      I$$RDN= 000000      L$NTFS  000102          R$NXT   000000      T$CFLG  000000
D$VBL   000032      K$HIC   000004      L$NTPK  000062          R$OWN   000010      T$CIN   000012
D$$BUG= 177514      K$LEN   000006      L$OMST  000002          R$PAL   000052      T$CLEN  000020
D$$ISK= 000000      K$LNK   000000      L$PCS   000112          R$PAS   000053      T$CMAP  000002
D$$L11= 000001      K$LOC   000002      L$PLS   000110          R$R0  = 000002      T$COUT  000014
D$$YNC= 000000      K$$CNT= 177546      L$QCNT  000122          R$R1  = 000004      T$CSIZ  000010
D$$YNM= 000000      K$$CSR= 177546      L$QUEH  000124          R$R2  = 000006      T$CTCB  000016
E$$XPR= 000000      K$$LDC= 000000      L$QUET  000130          R$R3  = 000010      T$CUSE  000001
F$$LVL= 000001      K$$TPS= 000074      L$RTRY  000007          R$R4  = 000012      T$LCHK  000076
F.RSIZ= ****** GX   LD$LP = 000000      L$SLN   000004          R$R5  = 000014      T$LFLG  000034
F.URBD= ****** GX   LN$OFF= 000001      L$ST    000005          R$USL   000031      T$LIDN  000030
GF$BUG= 000001      LN$ON = 000000      L$TCLZ  000044          R$USR   000032      T$LLEN  000100
G$CUG   000010      LN$SHU= 000002      L$TIM   000006          R$$DER= 000000      T$LTIM  000004
G$DTE   000012      LP$EIP= 002000      L$$ASG= 000000          R$$K11= 000001      T$LTRA  000000
G$FLG   000014      LP$ENB= 004000      L$$DRV= 000000          R$$SND= 000000      T$LTSK  000024
G$GNAM  000016      LP$PCT= 001400      L$$P11= 000001          R$$11M= 000000      T$RCNT  000002
G$LEN   000016      LP$TMR= 100000      L$$11R= 000000          SF$BLK= 000001      T$RFLG  000000
```

```
Symbol table
```

```
                                        ;+
                                        ; *** - MANINI - INITIALIZE MANAGEMENT FIELDS OF SLT
                                        ;
                                        ; INPUTS:
                                        ;       R0 = SLT ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       R1,R2 = DESTROYED
                                        ;
                                        ;-

 000512                                 MANINI:
 000512  016702  000000G                        MOV     $SLTNM,R2               ; GET NUMBER OF LINES IN SYSTEM
 000516  016701  000000G                        MOV     $SLTMA,R1               ; POINT TO SLT MAPPING TABLE
 000522  022100                         10$:    CMP     (R1)+,R0                ; FIND SLT POINTER IN TABLE
 000524  001403                                 BEQ     20$                     ; IF EQ, GOT IT
 000526  005302                                 DEC     R2                      ; ANY MORE SLTS ?
 000530  001374                                 BNE     10$                     ; IF NE, YES
 000532  000004                                 IOT                             ; SLT NOT FOUND ???
 000534  005741                         20$:    TST     -(R1)                   ; BACK UP TO SLT ADDRESS
 000536  166701  000000G                        SUB     $SLTMA,R1               ; CALCULATE SLN*2
 000542  066701  000000G                        ADD     $LLCTA,R1               ; POINT AT ENTRY IN LLC TABLE
 000546  116002  000000G                        MOVB    L.NSTA(R0),R2           ; GET NUMBER OF TRIBUTARIES
 000552  001424                                 BEQ     50$                     ; IF EQ, PT-TO-PT LINE

                                        ;
                                        ; MULTI-POINT LINE
                                        ;
 000554  010246                                 MOV     R2,-(SP)                ; SAVE NUMBER OF TRIBUTARIES
 000556  010002                                 MOV     R0,R2                   ; POINT TO START OF TRIBUTARIES
 000560  062702  000000G                        ADD     #L.MPF,R2               ; ...
 000564  011101                                 MOV     (R1),R1                 ; GET ADD/2 OF MULTI-POINT TABLE
 000566  100401                                 BMI     30$                     ; MAKE SURE SIGN BIT IS SET
 000570  000004                                 IOT                             ; CONSISTENCY ERROR
 000572  006301                         30$:    ASL     R1                      ; FORM ADDRESS OF TABLE
 000574  105721                         40$:    TSTB    (R1)+                   ; SKIP LLN
 000576  112162  000000G                        MOVB    (R1)+,S.OWNR(R2)        ; FILL IN MANAGEMENT OWNER
 000602  112762  000000G 000000G                MOVB    #LN.OFF,S.NMST(R2)      ; AND INITIAL STATE IS OFF
 000610  062702  000000G                        ADD     #S.LEN,R2               ; SKIP TO NEXT TRIBUTARY
 000614  005316                                 DEC     (SP)                    ; ANY MORE ?
 000616  003366                                 BGT     40$                     ; IF GT, YES
 000620  005726                                 TST     (SP)+                   ; ELSE, CLEAN UP STACK
 000622  000406                                 BR      60$                     ; AND RETURN

                                        ;
                                        ; PT-TO-PT LINE
                                        ;
 000624  116160  000001  000000G        50$:    MOVB    1(R1),L.OWNR(R0)        ; FILL IN OWNER
 000632  112760  000000G 000000G                MOVB    #LN.OFF,L.NMST(R0)      ; AND OFF STATE
 000640                                 60$:    RETURN

         000001                                 .END
```

Symbol table

B 6

```
**FILE**ID**VXLB









VV          VV    XX          XX    LL              BBBBBBBB
VV          VV    XX          XX    LL              BBBBBBBB
VV          VV    XX          XX    LL              BB        BB
VV          VV    XX          XX    LL              BB        BB
VV          VV      XX    XX        LL              BB        BB
VV          VV      XX    XX        LL              BB        BB
VV          VV        XX            LL              BBBBBBBB
VV          VV        XX            LL              BBBBBBBB
VV          VV      XX    XX        LL              BB        BB
VV          VV      XX    XX        LL              BB        BB
  VV      VV      XX          XX    LL              BB        BB        ....
  VV      VV      XX          XX    LL              BB        BB        ....
    VV  VV        XX          XX    LLLLLLLLLL      BBBBBBBB            ....
    VV  VV        XX          XX    LLLLLLLLLL      BBBBBBBB            ....






LL                SSSSSSSS    TTTTTTTTTT
LL                SSSSSSSS    TTTTTTTTTT
LL              SS                TT
LL              SS                TT
LL              SS                TT
LL              SS                TT
LL                SSSSSS          TT
LL                SSSSSS          TT
LL                      SS        TT
LL                      SS        TT
LL                      SS        TT
LL                      SS        TT
LLLLLLLLLL      SSSSSSSS          TT
LLLLLLLLLL      SSSSSSSS          TT
```

C 6

```
VXLB - VNP LIBRARY DEALLOCATION MACRO V05.03b Friday 06-Sep-85 14:39
Table of contents
```

```
                                                                B  7
VXPR - VNP RELEASE PROCFSS FROM MACRO V05.03b Saturday 29-Jun-85 03:12  Page 5-2
Symbol table

AS.DEL= 000010                   FE.X25= 004000                   HF.EIS= 000002                   N$$MOV= 000010                   P.TCB   000026
AS.EXT= 000004                   F$$LVL= 000001                   HF.FPP= 100000                   N$$NCT= 000001                   P.WAIT  000020
AS.RED= 000001                   F.NRBD= ****** GX                HF.QB = 000004                   N$$PEM= 000001                   Q$$OPT= 000010
AS.WRT= 000002                   F.RSIZ= ****** GX                HF.UBM= 000001                   PIRQ  = 177772                   R$$DER= 000000
A$$CHK= 000000                   F.URBD= ****** GX                I$$RAR= 000000                   PMODE = 030000                   R$$K11= 000001
A$$CPS= 000000                   F2.ACN= 000020                   I$$RDN= 000000                   PR0   = 000000                   R$$SND= 000000
A$$PRI= 000000                   F2.AHR= 010000                   KDSAR0= 172360                   PR1   = 000040                   R$$11M= 000000
A$$TRP= 000000                   F2.DAS= 000001                   KDSDR0= 172320                   PR4   = 000200                   SISDR0= 172200
A.IOC   000003                   F2.DPR= 000400                   KINAR0= 172340                   PR5   = 000240                   SR0   = 177572
A.LGTH= 000014                   F2.EVT= 000010                   KINAR5= 172352                   PR6   = 000300                   SR3   = 172516
A.MPCT  000011                   F2.GGF= 002000                   KINAR6= 172354                   PR7   = 000340                   SWR   = 177570
A.PCB   000012                   F2.IRR= 001000                   KINAR7= 172356                   PS    = 177776                   SYSFDB= ****** GX
A.PCBL  000000                   F2.LIB= 000002                   KISAR0= 172340                   PS.APR= 000007                   S$$WRG= 000000
A.PRI   000002                   F2.MP = 000004                   KISAR5= 172352                   PS.CHK= 010000                   S$$YSZ= 007600
A.STAT  000010                   F2.POL= 000100                   KISAR6= 172354                   PS.CKP= 040000                   TPS   = 177564
A.TCB   000004                   F2.RAS= 004000                   KISAR7= 172356                   PS.CKR= 020000                   T$$KMG= 000000
A.TCBL  000006                   F2.RBN= 020000                   KISDR0= 172300                   PS.COM= 000200                   T$$MIN= 000000
CMODE = 140000                   F2.SDW= 000040                   KISDR6= 172314                   PS.DEL= 000010                   UBMPR = 170200
C$$CKP= 000000                   F2.STP= 100000                   KISDR7= 172316                   PS.DRV= 000020                   UDSAR0= 177660
C$$ORE= 000400                   F2.SWP= 040000                   K$$CNT= 177546                   PS.FXD= 004000                   UDSDR0= 177620
C$$RSH= 177564                   F2.WND= 000200                   K$$CSR= 177546                   PS.LIO= 001000                   UISAR0= 177640
D$$BUG= 177514                   F3.AST= 000200                   K$$LDC= 000000                   PS.NSF= 000400                   UISAR4= 177650
D$$ISK= 000000                   F3.CLI= 001000                   K$$TPS= 000074                   PS.OUT= 100000                   UISAR5= 177652
D$$L11= 000001                   F3.CRA= 000001                   LD$LP = 000000                   PS.PER= 002000                   UISAR6= 177654
D$$YNC= 000000                   F3.EIS= 000004                   L$$ASG= 000000                   PS.PIC= 000100                   UISAR7= 177656
D$$YNM= 000000                   F3.NWK= 000002                   L$$DRV= 000000                   PS.SYS= 000040                   UISDR0= 177600
E$$XPR= 000000                   F3.PMN= 004000                   L$$P11= 000001                   P$$P45= 000000                   UISDR4= 177610
FE.CAL= 000040                   F3.PRO= 000040                   L$$11R= 000000                   P$$WRD= 000000                   UISDR5= 177612
FE.CEX= 020000                   F3.RLK= 020000                   MPAR  = 172100                   P.BLKS  000016                   UISDR6= 177614
FE.DRV= 000010                   F3.SHF= 040000                   MPCSR = 177746                   P.BUSY  000024                   UISDR7= 177616
FE.DYM= 010000                   F3.STM= 000010                   M$$CRB= 000124                   P.IOC   000003                   V$$CTR= 001000
FE.EXP= 000200                   F3.TCM= 002000                   M$$CRX= 000000                   P.LNK   000000                   X$$DBT= 000000
FE.EXT= 000001                   F3.UDS= 000020                   M$$FCS= 000000                   P.MAIN  000012                   $BFR  = ****** GX
FE.EXV= 000004                   F3.WAT= 010000                   M$$MGE= 000000                   P.NAM   000004                   $DEA16= ****** GX
FE.FDT= 002000                   F3.XHR= 000100                   M$$NET= 000000                   P.OWN   000026                   $FMASK= ****** GX
FE.LSI= 000400                   F3.11S= 000400                   M$$OVR= 000000                   P.PRI   000002                   $GETRV= ****** GX
FE.MUP= 000002                   G$$TPP= 000000                   N$$ACC= 000001                   P.REL   000014                   $PUTRC= ****** GX
FE.MXT= 040000                   G$$TSS= 000000                   N$$BUF= 000001                   P.SIZE  000016                   $RADDR= ****** GX
FE.NLG= 100000                   G$$TTK= 000000                   N$$LDV= 000001                   P.STAT  000030                   $XPAR   000000RG
FE.OFF= 001000                   G$$WRD= 000000                   N$$MCP= 000001                   P.SUB   000010                   .PLGTH= ****** GX
FE.PKT= 000100                   HF.CIS= 000200                   N$$MLL= 000001                   P.SWSZ  000022                   .PPRO = ****** GX
FE.PLA= 000020

. ABS.  000032    000   (RW,I,GBL,ABS,OVR)
        000244    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10635 Words  ( 42 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:08.52
SY:VXPR.V2,[132,134]VXPR/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VXPR
```

```
                                                          C  7
VXPR        CREATED BY  MACRO  ON 29-JUN-85 AT 03:12     PAGE 1

SYMBOL CROSS REFERENCE                                   CREF   04.00
```

Symbol table

```
A$$CHK= 000000          CS.DIS= 000040          D$END = 000104          FS.EXI= 001000          LF.SER= 000040
A$$CPS= 000000          CS.ENA= 000001          D$FNB   000034          FS.GET= 006000          LF.TIM= 000010
A$$PRI= 000000          CS.ENB= 000020          D$HIOR  000024          FS.HLT= 000000          LF.UNL= 020000
A$$TRP= 000000          CS.ERR= 100000          D$HOST  000022          FS.INI= 000000          LF.X2P= 000000
BUFUMP= 172354          CS.FTL= 001000          D$INAC  000044          FS.KIL= 000000          LLCAL = ****** GX
CB.CCB= 000002          CS.HCR= 000001          D$INCT  000042          FS.LCL= 100000          LN.CLO= 000000
CB.DDM= 000040          CS.HFE= 002000          D$IPL   000051          FS.LTM= 001000          LN.DUM= 000005
CB.DLC= 000020          CS.LST= 040000          D$LID   000020          FS.MNT= 004000          LN.LOA= 000004
CB.RDB= 000004          CS.MTL= 004000          D$LNAM  000006          FS.MSN= 014000          LN.LOO= 000003
CB.SPB= 000010          CS.RNG= 000010          D$LNUM  000014          FS.REA= 001000          LN.OAU= 000003
CB.SLI= 000100          CS.ROV= 000004          D$LST   000047          FS.RET= 000000          LN.OFF= 000001
CB.XLB= 000001          CS.RSN= 010000          D$MAXC  000064          FS.REZ= 003000          LN.ON = 000000
CC.LLC= 000200          CS.SHU= 000001          D$MAXH  000066          FS.RLB= 002000          LN.OOP= 000004
CE.ABO= 100362          CS.SID= 000002          D$MAXV  000070          FS.RNG= 011000          LN.OPE= 000001
CE.DAO= 100346          CS.STR= 000004          D$MLL   000040          FS.RST= 000000          LN.REF= 000002
CE.DIS= 100366          CS.SUC= 000001          D$MNOD  000041          FS.RTN= 001000          LN.SER= 000002
CE.ERR= 100370          CS.TMO= 020000          D$NA    000062          FS.SET= 005000          LN.STA= 000017
CE.ILN= 100350          CS.XUR= 000004          D$NBEA  000056          FS.SFC= 005000          LN.SUB= 000360
CE.LTO= 100356          C$$CKP= 000000          D$NBRA  000054          FS.SFR= 006000          LN.TRI= 000006
CE.MOP= 100372          C$$ORE= 000400          D$NEND= 000054          FS.SFS= 004000          LS.CEX= ****** GX
CE.NTE= 100361          C$$RSH= 177564          D$NLN   000030          FS.SPW= 040000          LS.NTI= ****** GX
CE.RTE= 100376          C.ADD   000034          D$NN    000060          FS.STM= 000000          LS.TOP= ****** GX
CE.SRC= 100364          C.BID   000003          D$OUTT  000043          FS.STP= 002000          LX.CEX= ****** GX
CE.STP= 100352          C.BUF   000014          D$RETF  000050          FS.STR= 001000          L$$ASG= 000000
CE.TME= 100354          C.BUF1  000014          D$RNN   000002          FS.TRM= 003000          L$$DRV= 000000
CE.TMO= 100374          C.BUF2  000024          D$RTMR  000076          FS.WLB= 001000          L$$P11= 000001
CE.UNS= 100344          C.CNT   000020          D$SEG   000036          FS.XKL= 002000          L$$11R= 000000
CF.CHN= 000001          C.CNT1  000020          D$SER   000032          FS.XOF= 010000          L.COST  000015
CF.EOM= 000004          C.CNT2  000030          D$SQRL  000052          FS.XON= 007000          L.CTL   000012
CF.HDR= 000020          C.FLG   000022          D$$BUG= 177514          FS.ZER= 002000          L.CVA   177776
CF.LB = 100000          C.FLG1  000022          D$$ISK= 000000          F$$LVL= 000001          L.DDM   000002
CF.LIN= 000002          C.FLG2  000032          D$$L11= 000001          G$$TPP= 000000          L.DDS   000004
CF.SOM= 000010          C.FNC   000010          D$$YNC= 000000          G$$TSS= 000000          L.DLC   000003
CF.SYN= 000040          C.LIN   000006          D$$YNM= 000000          G$$TTK= 000000          L.DLM   000006
CF.TRN= 000100          C.LNK   000000          E$$XPR= 000000          G$$WRD= 000000          L.DLS   000010
CMODE = 140000          C.MOD   000011          FC.CCP= 000020          INIFUN  000030R     002 L.FLG   000000
CM.CIR= 000002          C.NSP   000004          FC.CTL= 000006          I$$RAR= 000000          L.KRBA  000016
CM.FMT= 100000          C.PRO   000042          FC.KCP= 000016          I$$RDN= 000000          L.LEN = 000022
CM.HRD= 000002          C.RSV   000002          FC.KIL= 000004          KISAR0= 172340          L.MPF   000022
CM.LIN= 000000          C.STA   000007          FC.MAN= 000024          KISAR6= 172354          L.NMST  000020
CM.LOO= 000001          C.STS   000012          FC.MLD= 000026          KSAR6 = ****** GX       L.NSTA  000014
CM.XLO= 000004          C.URM   177776          FC.PCT= 000030          K$$CNT= 177546          L.OWNR  000021
CP.DCF= 000040          C.XACP  000004          FC.PWR= 000022          K$$CSR= 177546          L.UNT   000013
CP.HDL= 000007          C.XID   000035          FC.RCE= 000002          K$$LDC= 000000          MPAR  = 172100
CP.PS = 177400          C.XLEN  000044          FC.RCP= 000014          K$$TPS= 000074          MPCSR = 177746
CP.PSI= 000200          C.XPLI  000040          FC.TIM= 000010          LD$LP = 000000          M$$CRB= 000124
CP.XCF= 000100          C.XPT   000034          FC.XCP= 000012          LF.ACT= 100000          M$$CRX= 000000
CP.2FR= 000030          C.XSVC  000042          FC.XME= 000000          LF.BRO= 000400          M$$EXT= 000000 G
CS.ABO= 000100          C.XTC   000037          FE.CEX= ****** GX       LF.BWT= 000007          M$$FCS= 000000
CS.BRO= 000002          C.X25   000036          FMASK = ****** GX       LF.ENA= 002000          M$$MGE= 000000
CS.BUF= 000200          D$AMXC  000072          FNDPDV= ****** GX       LF.LPB= 001000          M$$NET= 000000
CS.CES= 000002          D$AMXH  000074          FS.AST= 000000          LF.MDC= 000100          M$$OVR= 000000
CS.CHN= 000010          D$ANN   000000          FS.CIB= 002000          LF.MFL= 004000          NTILIN  000110RG
CS.CMP= 000200          D$BRPR  000102          FS.CRA= 001000          LF.MTP= 000020          NTLPT = ****** GX
CS.DCR= 000400          D$BRTM  000100          FS.DIS= 013000          LF.PAC= 000200          N$$ACC= 000001
CS.DEF= 000004          D$DELF  000045          FS.DVC= 001000          LF.RDY= 040000          N$$BUF= 000001
CS.DEV= 000002          D$DELW  000046          FS.ENB= 012000          LF.REA= 010000          N$$LDV= 000001
```

Symbol table

```
                                                                                  B  9
NTICSR      CREATED BY  MACRO  ON 29-JUN-85 AT 03:25       PAGE 1
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL   VALUE               REFERENCES

BUFUMP  = 172354             #5-61
CMODE   = 140000             #5-61
CSRDIS    000006 R           #5-75         6-151
CSRENA    000000 R           #5-74         6-149
CSRSLT    000016 R           #5-82        *6-134         6-157
CSRTAB    000014 R            5-74        #5-77
CSRTRP    000274 R            5-77        #6-204
I$$AS   = ******              5-61         5-63
KISAR0  = 172340             #5-61
KISAR6  = 172354             #5-61
KSAR6   = ******    GX        6-124        6-129
LF.ACT  = 100000             #5-62
LF.BRO  = 000400             #5-62
LF.BWT  = 000007             #5-62
LF.ENA  = 002000             #5-62
LF.LPB  = 001000             #5-62
LF.MDC  = 000100             #5-62
LF.MFL  = 004000             #5-62
LF.MTP  = 000020             #5-62
LF.PAC  = 000200             #5-62
LF.RDY  = 040000             #5-62
LF.REA  = 010000             #5-62
LF.SER  = 000040             #5-62
LF.TIM  = 000010             #5-62
LF.UNL  = 020000             #5-62
LF.X2P  = 000000             #5-62
LN.CLO  = 000000             #5-62
LN.DUM  = 000005             #5-62
LN.LOA  = 000004             #5-62
LN.LOO  = 000003             #5-62
LN.OAU  = 000003             #5-62
LN.OFF  = 000001             #5-62
LN.ON   = 000000             #5-62
LN.OOP  = 000004             #5-62
LN.OPE  = 000001             #5-62
LN.REF  = 000002             #5-62
LN.SER  = 000002             #5-62
LN.STA  = 000017             #5-62
LN.SUB  = 000360             #5-62
LN.TRI  = 000006             #5-62
LR.CTL  = ******    GX       *6-176
LR.LIN  = ******    GX       *6-178
LR.STS  = ******    GX       *6-175
LR.UNT  = ******    GX       *6-177
LX.LIN  = ******    GX        6-175
L.COST    000015             #5-62
L.CTL     000012             #5-62         6-176         6-186
L.CVA     177776             #5-62
L.DDM     000002             #5-62         6-162         6-187
L.DDS     000004             #5-62
L.DLC     000003             #5-62
L.DLM     000006             #5-62
```

```
                                                                                  C  9
NTICSR      CREATED BY  MACRO  ON 29-JUN-85 AT 03:25       PAGE 2
SYMBOL CROSS REFERENCE                                     CREF    04.00
```

```
    295 000566                        350$:    EMSG$R  17                         ;RUN$ FAILURE
    296 000574                        360$:    EMSG$R  18                         ;PLB ALLOCATION FAILURE
    297 000602                        370$:    EMSG$R  11                         ;ROUTER PRIORITIES DATA BASE ALLOC. FAILURE
    298
```

```
                                        .TITLE  NTIMFE - DO MARKED-FOR-ENABLE SCAN
                                        .IDENT  /V05.00/

                                ;
                                ; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ;       NTINIT - MARKED-FOR-ENABLE SCAN
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  27-FEB-78
                                ;       VERSION 2.0 RELEASE
                                ;
                                ; 2.00  14-DEC-79
                                ;       DECNET-11M/S V3.0
                                ;       DECNET-11M-PLUS V1.0
                                ;
                                ; 3.00  16-APR-82
                                ;       DECNET-11M V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;

```

B 12

```
NTIMFE      CREATED BY  MACRO  ON 24-JUL-85 AT 15:09      PAGE 3
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE             REFERENCES

SF.MFL  = 000040           #5-74
SF.PAC  = 000020           #5-74
SF.REA  = 000010           #5-74
SF.SER  = 000001           #5-74
SF.SVC  = 000002           #5-74        8-356
SF.UNL  = 000040           #5-74
SLTMA   = ******  GX        6-122
SLTNM   = ******  GX        6-120
STAADD    000000 R         #5-90       *6-124      *6-197       8-356
STANUM    000002 R         #5-91       *6-167      *6-205       8-368
STRT    = 000004           #5-84        7-295
S$CTL     000002            8-366
S$STA     000003            8-368
S$$BAS  = ******            6-112
S.COST    000001           #5-74
S.FLG     000000           #5-74       *6-192
S.LEN     000004           #5-74
S.NMST    000002           #5-74       *6-202
S.OWNR    000003           #5-74
TMEFN   = 000001           #5-85        6-113       6-114
T$FLAG    000044           #5-72       *8-355      *8-358
T$LIF     000013           #5-72
T$LIFL    000013           #5-72
T$LIFO    000013           #5-72
T$LIFS    000013           #5-72
T$LIN     000000           #5-72
T$LIPS    000006           #5-72
T$LLD     000012           #5-72
T$LLDC    000045           #5-72
T$LLDL    000012           #5-72
T$LLDO    000012           #5-72
T$LLDS    000012           #5-72
T$LLEN    000046           #5-72        8-352
T$LOPR    000002           #5-72
T$LTCL    000024           #5-72
T$LTIM    000026           #5-72
T$LTPR    000014           #5-72
T$LTPS    000020           #5-72
T$NAPL    000004           #5-72
T$NFE     000000           #5-72
T$NLEN    000010           #5-72
T$NNUL    000002           #5-72
T$NOPL    000006           #5-72
T$NRNI    000042           #5-72
T$NRPL    000005           #5-72
T$NRUL    000007           #5-72
T$NVR     000001           #5-72
T$RPRI    000040           #5-72
T$SVC     000034           #5-72       *8-372      *8-373
T$T5      000030           #5-72
T$T6      000032           #5-72
X$$MCB  = ******            5-77        5-77
```

C 12

```
NTIMFE      CREATED BY  MACRO  ON 24-JUL-85 AT 15:09      PAGE 4
SYMBOL CROSS REFERENCE                                    CREF   04.00
```

```
                                                                                B 13
NTIMFL      CREATED BY  MACRO  ON 29-JUN-85 AT 03:27      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

BUFUMP   = 172354           #5-66
CMODE    = 140000           #5-66
DLMMFL     000344 R          8-141       #9-187
1$$AS    = ******            5-64         5-66
KISAR0   = 172340           #5-66
KISAR6   = 172354           #5-66
LF.ACT   = 100000           #5-65
LF.BRO   = 000400           #5-65
LF.BWT   = 000007           #5-65
LF.ENA   = 002000           #5-65
LF.LPB   = 001000           #5-65
LF.MDC   = 000100           #5-65
LF.MFL   = 004000           #5-65         8-134
LF.MTP   = 000020           #5-65
LF.PAC   = 000200           #5-65
LF.RDY   = 040000           #5-65
LF.REA   = 010000           #5-65
LF.SER   = 000040           #5-65
LF.TIM   = 000010           #5-65
LF.UNL   = 020000           #5-65
LF.X2P   = 000000           #5-65
LN.CLO   = 000000           #5-65
LN.DUM   = 000005           #5-65
LN.LOA   = 000004           #5-65
LN.LOO   = 000003           #5-65
LN.OAU   = 000003           #5-65
LN.OFF   = 000001           #5-65
LN.ON    = 000000           #5-65
LN.OOP   = 000004           #5-65
LN.OPE   = 000001           #5-65
LN.REF   = 000002           #5-65
LN.SER   = 000002           #5-65
LN.STA   = 000017           #5-65
LN.SUB   = 000360           #5-65
LN.TRI   = 000006           #5-65
LO.CON     000006           #5-67
LO.CSR     000006           #5-67
LO.CTL     000014           #5-67        *9-200
LO.LIN     000004           #5-67
LO.PAR     000010           #5-67
LO.PRI     000010           #5-67
LO.SCR     000014           #5-67
LO.STA     000015           #5-67        *9-199
LO.VCT     000012           #5-67
LR.AST     000026           #5-67
LR.CIR     000002           #5-67        *9-194
LR.CTL     000004           #5-67        *8-148        8-158
LR.EFN     000004           #5-67
LR.LIN     000002           #5-67        *8-150        8-156
LR.PRM     000030           #5-67
LR.PRO     000002           #5-67        *7-121        7-126
LR.STS     000000           #5-67        *7-117       *7-120       *8-144       *8-147       *9-195       *9-198
```

```
                                                                                C 13
NTIMFL      CREATED BY  MACRO  ON 29-JUN-85 AT 03:27      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF    04.00
```

```
                                          .SBTTL NETWORK INITIALIZATION COLD START

                                  ;+
                                  ; NTINIT - ENTER THE NTINIT TASK
                                  ;
                                  ; THIS ENTRY SHOULD ONLY BE TAKEN ONCE, AS IT WILL STAY ACTIVE,
                                  ; WAITING FOR MORE INSTRUCTIONS VIA "SEND DATA".
                                  ;
                                  ; INPUTS:
                                  ;         NTINIT GET INPUT VIA "RCV DATA" AND RETURNS COMPLETION VIA
                                  ;           "SEND DATA"
                                  ;         THE FIRST WORD OF THE RCV DATA PACKET ON RCV INDICATES THE
                                  ;           FUNCTION TO BE PERFORMED
                                  ;         THE FIRST WORD OF THE SEND DATA PACKET ON COMPLETION INDICATES
                                  ;           SUCCESS (+), FAILURE (-), OR NO-OP (0)
                                  ;         WHEN SENDING A FUNCTION REQUEST TO NTINIT, THE REQUEST SHOULD
                                  ;           BE QUEUED FIRST (VIA SEND DATA TO NTINIT) AND NTINIT SHOULD
                                  ;           THEN BE REQUESTED
                                  ;         EVENT FLAGS START OFF "CLEAR"
                                  ;
                                  ; OUTPUTS:
                                  ;         CHECK COMPATIBILITY OF RSX SYSTEM
                                  ;         START MAIN CONTROL LOOP
                                  ;
                                  ;-

000000                            NTINIT:
                                          .IF DF   R$$MPL
                                          .IF NDF  R$$PRO
                                          DIR$     #TRNVEC                    ; INITIALIZE VECTOR TABLE
                                          BCC      1$                         ; IF SUCCESS - BRANCH
                                          IOT                                 ; THIS SOULD NEVER OCCUR
                                  1$:     BIT      #FE.CEX,@FMASK             ; COMM EXEC LOADED ?
                                          BEQ      5$                         ; IF NOT - BRANCH
                                          CALL     VECSET                     ; IF YES - INITIALIZE PROCESS VECTOR TABLES
                                  5$:
                                          .ENDC
                                          .ENDC

                                          .IF NDF  R$$11S
000000                                    CLEF$S   #15.                       ;CLEAR EVENT FLAG
000012                                    CALL     NTIRDX                     ;GET FIRST RECEIVE DATA OR EXIT
000016                                    SRDA$S   #NTIRDA                    ;SET THE RCV DATA AST ENTRY

000030  005067  000120'                   CLR      $NIAST                     ;ASSUME CALLER NOT NOTIFIED VIA AST
000034  032767  100000   000204'          BIT      #LS.AST,$NIFNC             ;NOTIFY CALLER VIA AST?
000042  001406                            BEQ      10$                        ;BR IF NO
000044  016767  000060'  000120'          MOV      LR.AST+$NIRDB,$NIAST       ;STORE ADDRESS OF NOTIFICATION AST
000052  016767  000062'  000122'          MOV      LR.PRM+$NIRDB,$NIPRM       ;STORE NOTIFICATION AST PARAMETER
000060                            10$:
                                          .IFF
                                          DIR$     #NTIRCX                    ;RECEIVE DATA OR EXIT
                                          .ENDC

                                          .IF DF  S$$BAS
                                                                              ; FOR THE SERVER BASE, WE NEED TO
                                                                              ; DO SOME OF THE THINGS WE WOULD
                                                                              ; DO FOR 11M - THIS IS CODE COPIED FROM
```

```
                                        .SBTTL  RCV DATA SUBROUTINE

                                ;+
                                ; NTIRDX - RCV DATA AND RETURN OR EXIT
                                ; NTIRDT - PROCESS THE RECEIVED DATA
                                ;
                                ; THESE ROUTINES GET A RCV DATA PACKET FROM EITHER A FUNCTION REQUESTOR
                                ; OR FROM NTL.
                                ;
                                ; INPUTS:
                                ;       LOW BYTE OF $NIFLG IS EITHER ZERO (RCV FUNCTION REQUEST) OR
                                ;         NON-ZERO (BUSY WITH FUNCTION REQUEST, RCV ONLY FROM NTL)
                                ;
                                ; OUTPUTS:
                                ;       THE DATA IS RECEIVED INTO "$NIRTN,$NIRDB".
                                ;       IF THE LOW BYTE OF $NIFLG IS ZERO, THE SENDERS NAME IS SAVED
                                ;         IN THE "NTISEN" DIRECTIVE, THE FUNCTION CODE IS SAVED
                                ;         IN "$NIFNC".
                                ;       IF THE LOW BYTE OF $NIFLG IS NON-ZERO, AND THE SENDER IS NTL,
                                ;         NTL'S ANSWER IS SAVED IN "$NTLFC" AND EVENT FLAG 15. IS
                                ;         SET, ELSE A "BUSY" RESPONSE IS SENT TO THE REQUESTOR.
                                ;-
                                        .IF NDF R$$11S

001072                          NTIRDX: DIR$    #NTIRCX                 ;RECEIVE DATA OR EXIT

001100  012700  000026'         NTIRDT: MOV     #$NIRTN,R0              ;POINT AT SENDING TASK'S NAME
001104  105767  000226'                 TSTB    $NIFLG                  ;ARE WE ALREADY ACTIVE ?
001110  001427                          BEQ     30$                     ;BR IF NO, RCV FUNCTION REQUEST

001112  022720  055254                  CMP     #^RNTL,(R0)+            ;IS DATA FROM NTL ?
001116  001014                          BNE     20$                     ;BR IF NO, ERROR
001120  022720  131574                  CMP     #^R...,(R0)+            ;MAYBE ?
001124  001010                          BNE     10$                     ;BR IF NO, STILL AN ERROR
001126  011067  000164'                 MOV     (R0),$NTLFC             ;SAVE NTL'S ANSWER
001132                                  SETF$S  #15.                    ;SET THE EVENT FLAG
001144  000425                          BR      40$                     ;AND LEAVE

001146  005740                  10$:    TST     -(R0)                   ;RESET THE POINTER
001150  011067  000216'         20$:    MOV     (R0),NTIBSY+S.DATN+2    ;SET THE REQUESTORS TASK NAME INTO
001154  014067  000214'                 MOV     -(R0),NTIBSY+S.DATN     ;  THE BUSY RESPONSE DIRECTIVE
001160                                  DIR$    #NTIBSY                 ;TELL REQUESTOR WE ARE BUSY
001166  000414                          BR      40$                     ;EXIT WITH CARRY CLEAR

001170                          30$:                                    ; (NOTE THE FOLLOWING 5 LINES ARE
                                                                        ; DUPLICATED IN SERVER BASE CODE
                                                                        ; AFTER NTINIT - PLEASE MAKE CHANGES
                                                                        ; TO BOTH PLACES)
001170  012067  000174                  MOV     (R0)+,NTISEN+S.DATN     ;SAVE SENDER'S NAME IN
001174  012067  000176'                 MOV     (R0)+,NTISEN+S.DATN+2   ;  RETURN DIRECTIVE
001200  012067  000204'                 MOV     (R0)+,$NIFNC            ;SAVE FUNCTION CODE
001204  012067  000206'                 MOV     (R0)+,$NIFNC+2          ; SAVE TWO WORDS OF SENDER'S DATA
001210  011067  000210'                 MOV     (R0),$NIFNC+4           ; (USED FOR POWERFAIL RECOVERY)
001214  105267  000226'                 INCB    $NIFLG                  ;WE ARE NOW BUSY

001220                          40$:    RETURN

```

B 16

```
NTINIT      CREATED BY  MACRO  ON 5-SEP-85 AT 11:04      PAGE 8
MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME      REFERENCES

ASTX$S          #5-59     15-768
CALL             8-299     9-378     9-380     9-387     9-404     9-428     9-443     9-447    12-590    12-592
                 12-602    13-681    14-716    14-726    14-737    15-765    17-853    17-855    17-859    17-875
CALLR           # -61      9-449    14-730    14-742
CALLX           #5-61
CCBDF$          #5-61      5-73
CLEF$S          #5-59     #5-60      8-298    11-561
DIR$            #5-60     #8-298     8-298    #8-300     8-300     9-401    #9-402     9-402     9-403     9-425
                #9-426     9-426     9-427    #9-439     9-439     9-446    #9-448     9-448    11-557    11-559
                #11-560   11-560   #11-561    11-561   #13-689    13-689    15-763   #15-768    15-768    16-796
                #16-807   16-807    16-813
DSAR$S          #5-60      9-439
EMSG$           #5-59      9-378     9-380
EMSG$R          #5-62     11-564
ENAR$S          #5-60      9-448
EXIT$S          #5-59
MAP             #5-61
MOV$            #8-298     8-298    #8-300     8-300    #9-402     9-402    #9-426     9-426   #11-560    11-560
                #11-561   11-561   #13-689    13-689    13-689    13-689   #13-689    13-689    13-689   #13-689
                 13-689   13-689   #16-807    16-807
MRKT$           #5-62      6-109
MVB$            #13-689   13-689
NHWJF$          #5-61      5-77
NTLDF$          #5-61      5-72
NTLER$          #5-59      7-190     7-192     7-193     7-194
OFF$            #6-104    #6-109    #6-110    #6-116    #6-118    #6-119    #6-122    #6-124
OPTDF$          #5-61      5-74
PCBDF$          #5-63      5-78
PDVDF$          #5-61      5-75
QDPB$S          #13-689   13-689
QIOW$S          #5-59     13-689
RCVD$           #5-60      6-116
RCVX$           #5-62      6-104
RESMAP          #5-63
RETC            #5-62     17-862
RETURN           11-562   12-611    13-697    14-720    14-743    16-827    17-876
RQST$           #5-60      6-110     6-118
RVP$            #13-689   13-689    13-689    13-689
R50$            #6-104     6-104    #6-110     6-110    #6-116     6-116    #6-118     6-118    #6-119     6-119
                #6-122     6-122    #6-124     6-124
SAVMAP          #5-63
SCBDF$          #5-63      5-79
SDAT$           #5-60      6-119     6-122     6-124
SDAT$S          #5-62
SETF$S          #5-60     16-807
SLTDF$          #5-61      5-76
SRDA$S          #5-60      8 ·300
SWSTK$           12-590   17-853
TCBDF$          #5-63      5-80
VECDF$          #5-62      5-81
WSIG$S          #5-59
WTSE$S          #5-59      9-402     9-426    11-560
```

C 16

```
**FILE**ID**NTITIM
```

```
004774  112102                 MOVB    (R1)+,R2          ;       LENGTH OF THIRD FIELD
004776  060201                 ADD     R2,R1             ;       ...
005000  162701  000000G        SUB     #$BFR,R1          ;       TOTAL LENGTH OF ALIAS BLOCK
005004                         CALL    $DEPOL            ; DE-ALLOCATE THE OBJECT BLOCK
005010  012600                 MOV     (SP)+,R0          ; GET UNMAPPED ADDRESS OF NEXT BLOCK
005012  001343                 BNE     10$               ; IF NE, MORE
005014  005067  000000G        CLR     $RBLCK            ; CLEAN UP
005020                 30$:    RETURN

                       ;
                       ; DE-ALLOCATE REMOTE TABLES
                       ;
005022                 XREMTB: SWTCHI  #$HMBUF           ; USE TEMPORARY BUFFER
005030                         GETRV   $DECPT,#D$END     ; READ DECNET HOME BLOCK
005050                         SWTCHI                    ; RESET DEFAULT BUFFER

005056  012703  000004G        MOV     #$HMBUF+D$RNN+2,R3 ; POINT AT START OF OBJECT TABLE LIST
005062  011300                 MOV     (R3),R0           ; SKIP LOCAL NODES BLOCK
005064  001430                 BEQ     30$               ; IF EQ, NO NODE NAME BLOCKS

005066                 10$:    CEACC$  R0,R3             ; CONVERT TO MAPPED ADDRESS

005076                 20$:    GETAD   R3,#4             ; GET THE REMOTE NODE NMAE BLOCK
005114  016746  000000G        MOV     $BFR,-(SP)        ; SAVE THE ADDRESS OF NEXT BLOCK
005120  032700  000001         BIT     #1,R0             ; REMOTE NODE BLOCK ALLOCATED FROM POOL.. ?
005124  001004                 BNE     25$               ; IF YES - DO NOT DEALLOCATE
005126  012701  000014         MOV     #R.LEN,R1         ; GET LENGTH OF REMOTE NODE NAME BLOCK
005132                         CALL    $DEPOL            ; DE-ALLOCATE THE REMOTE NODE NAME BLOCK
005136  012600         25$:    MOV     (SP)+,R0          ; POINT AT NEXT BUFFER
005140  001352                 BNE     10$               ; IF NE, MORE
005142  005067  000000G        CLR     $RBLCK            ; CLEAN UP
005146                 30$:    RETURN

                       ;
                       ; DE-ALLOCATE EVENT FILTER BLOCK TABLES
                       ;
005150                 XFLTTB:
005150  016700  000000G        MOV     $FILHD,R0         ; POINT AT START OF EVENT FILTER TABLE LIST
005154  001427                 BEQ     30$               ; IF EQ, END OF LIST

005156                 10$:    CEACC$  R0,R3             ; CONVERT TO MAPPED ADDRESS

005166                 20$:    GETAD   R3,#4             ; GET THE EVENT FILTER BLOCK
005204  016746  000000G        MOV     $BFR,-(SP)        ; SAVE THE ADDRESS OF NEXT BLOCK
005210  016701  000002G        MOV     F.CLS+$BFR,R1     ; GET LENGTH OF BLOCK (IN LOWER 6 BITS)
005214  042701  177700         BIC     #^C<FF.MSK>,R1    ; ISOLATE LENGTH
005220                 25$:    CALL    $DEPOL            ; DE-ALLOCATE THE FILTER BLOCK
005224  012600                 MOV     (SP)+,R0          ; POINT AT NEXT BUFFER
005226  001353                 BNE     10$               ; IF NE, MORE
005230  005067  000000G        CLR     $RBLCK            ; CLEAN UP
005234  005067  000000G 30$:   CLR     $LGSTT            ; CLEAR LOGGING STATE WORD
005240                         RETURN
```

```
VUNL
SYME
SYME
L.DL
L.FL
L.KF
M.LC
NSIT
NTIF
N$AD
N$AD
N$EL
N$EN
N$MH
N$MH
N$RC
N$RC
N$TL
N$XL
N$$V
O.LE
PRLT
PR.I
PR.S
PR7
P.NA
P.RE
P.SI
RF.I
R$$I
R$$I

R$$
R.AI
R.FI
R.LI
R.LI
R.N.
SCAI
SYS

TSK
T$F
T$L
T$L
T$L
T$L
.$L
T$L
T$L
T$L

VUNL
SYME
```

```
                                                                      C  2
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31      PAGE  3
SYMBOL  CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

L.DLS    = ******  GX        12-566    *12-577     14-762    *14-770    *19-1185
L.FLG    = ******  GX       *14-651    *14-778     15-863     21-1290
L.KRBA   = ******  GX        12-561    *12-564     14-690     14-748     17-960    *19-1182
M.LGTH   = 000014            21-1430
NSITAB     000130 R         #6-166      17-1032
NTIR       000124 R         #6-161      21-1261
N$ADJ1     000072            10-481
N$ADJ2     000074            10-500
N$ELEN     000054            11-524
N$ENC      000042            11-525     11-528     11-532
N$MHC1     000036            10-443     10-450
N$MHC2     000044            10-444
N$ROA1     000022            10-455     10-464
N$ROA2     000030            10-458
N$TLC      000100            10-468     10-477
N$XLEN     000124            10-442     10-453     10-466     10-479     10-497
N$$VCT   = ******            7-207      21-1494    21-1515    21-1545    21-1565
O.LEN      000012            21-1497
PRLTA      000200 R         #6-180     *17-1016   *17-1019   *17-1022    17-1023    18-1097    18-1103    23-1665   *23-1667
PR.LTA   = 000001           #6-182      17-1019    17-1023    18-1097    23-1665    23-1667
PR.SEC   = 000002           #6-183      17-1022    18-1103
PR7      = ******  GX        17-1038
P.NAM      000004            8-323      8-325      8-335      8-337      8-341      8-343
P.REL      000014            24-1818
P.SUB      000010            8-329
RF.LOO   = 100000           #5-82
R$$EIS   = ******            10-449     10-463     10-476     10-493     11-531     17-1031
R$$MPL   = ******            5-65       6-101      6-117      8-269      8-356      14-660     14-694     14-702     14-730
                             14-804     15-826     15-835     16-896     17-954     17-966     17-1006    17-1049    18-1092
                             18-1122    21-1274    21-1315    21-1326    21-1378    21-1396    21-1434    23-1671
R$$11D   = ******            10-449     10-463     10-476     10-493     11-531     17-1031
R.ADD      000010           #5-82
R.FLAG     000012           #5-82
R.LEN      000014           #5-82       21-1551
R.LNK      000000           #5-82
R.NAM      000002           #5-82
SCAN       002266 R          14-652    #15-857
SYSFDB   = ******  GX       *7-208     *8-322     *8-334     *10-442    *10-453    *10-466    *10-479    *10-485    *10-497
                            *11-524    *14-700    *15-827    *15-836    *17-962    *17-1007   *17-1047   *17-1050   *18-1089
                            *18-1165   *20-1225   *20-1226   *20-1227   *21-1264   *21-1271   *21-1394   *21-1469   *21-1496
                            *21-1508   *21-1509   *21-1510   *21-1517   *21-1537   *21-1538   *21-1539   *21-1547   *21-1567
                            *22-1596   *22-1641   *23-1773   *23-1776   *23-1778   *24-1814   *25-1849
TSKSCH     006260 R          21-1262   #25-1848
T$FLAG     000044           #5-85
T$LIF      000013           #5-85
T$LIFL     000013           #5-85
T$LIFO     000013           #5-85
T$LIFS     000013           #5-85
.$LIN      000000           #5-85
T$LIPS     000006           #5-85
T$LLD      000012           #5-85
T$LIDC     000045           #5-85
```

```
                                                                      D  2
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31      PAGE  4
SYMBOL  CROSS REFERENCE                                    CREF    04.00
```

```
                                              .SBTTL  $XPSI - Unload PSI data bases                          ;[TD001]
                                    ;+                                                                     ;[TD001]
                                    ; *** $XPSI - Unload psi data bases from extended pool                 ;[TD001]
                                    ;                                                                      ;[TD001]
                                    ; INPUTS:                                                              ;[TD001]
                                    ;         None.                                                        ;[TD001]
                                    ;                                                                      ;[TD001]
                                    ; OUTPUTS:                                                             ;[TD001]
                                    ;         PSI data bases are de-allocated                              ;[TD001]
                                    ;-                                                                     ;[TD001]
                                                                                                           ;[TD001]
000000                              $XPSI:: VNPDBG  VUNPSI          ; VNP debugging break point            ;[TD001]
000000  016703  000000G                     MOV     $PSIPT,R3       ; Get address of PSI home block        ;[TD001]
000004  001503                              BEQ     100$            ; Branch if none - nothing to do       ;[TD001]
000006                                      CALL    DEAPVC          ; Deallocate PVC name blocks           ;[TD001]
000012                                      CALL    DEACUG          ; Deallocate CUG name block            ;[TD001]
000016  062703  000004                      ADD     #H$RDTE,R3      ; Point to remote DTE name listhead    ;[TD001]
000022  012701  000024                      MOV     #R$LEN,R1       ; Get length of remote DTE name block  ;[TD001]
000026                                      CALL    DEALST          ; Deallocate remote DTE name block     ;[TD001]
000032  062703  000032                      ADD     #H$SVC-H$RDTE,R3 ; Point to SVC descriptor listhead    ;[TD001]
000036  012701  000032                      MOV     #S$LEN,R1       ; Get length of SVC descriptor block   ;[TD001]
000042                                      CALL    DEALST          ; Deallocate SVC descriptor blocks     ;[TD001]
000046  062703  177754                      ADD     #H$DST-H$SVC,R3 ; Point to X.25 destination descriptors ;[TD001]
000052                                      CALL    DEADST          ; Deallocate destination descriptors   ;[TD001]
000056  062703  000002                      ADD     #H$D29-H$DST,R3 ; Point to X.29 destination descriptors ;[TD001]
000062                                      CALL    DEADST          ; Deallocate destination descriptors   ;[TD001]
000066  032767  000000G 000000G             BIT     #PF.XGA,$PFLAG  ; Check for XGA module (host access)   ;[TD001]
000074  001406                              BEQ     10$             ; Branch if not present                ;[TD001]
000076  062703  000026                      ADD     #H$RNW-H$D29,R3 ; Point to remote network blocks listhea;[TD001]
000102  012701  000104                      MOV     #R$NWL,R1       ; Get length of remote network block   ;[TD001]
000106                                      CALL    DEALST          ; Deallocate remote network blocks     ;[TD001]
000112                              10$:    CALL    XPORT           ; Deallocate port table and circuit bloc;[TD001]
000116                                      CALL    XLDTE           ; Deallocate local DTE descriptor blocks;[TD001]
000122                                      CALL    XX29            ; Deallocate X.29 data block           ;[TD001]
000126                                      CALL    XTRA            ; Deallocate trace structures          ;[TD001]
000132                                      CALL    XPSV            ; Deallocate PSV mapping tables        ;[TD001]
000136  016700  000000G                     MOV     $PSIPT,R0       ; Get address of PSI home block        ;[TD001]
                                    ;                                                                      ;[TD001]
                                    ; Determine length of PSI homeblock and deallocate it                  ;[TD001]
                                    ;                                                                      ;[TD001]
000142  012701  000042                      MOV     #H$LEN,R1       ; Get length of old-style PSI home block;[TD001]
000146  032767  000000G 000000G             BIT     #PF.XDF,$PFLAG  ; Check for extended homeblock         ;[TD001]
000154  001007                              BNE     20$             ; Branch if extended                   ;[TD001]
000156  032767  000000G 000000G             BIT     #PF.XGA,$PFLAG  ; Check for host with old data formats ;[TD001]
000164  001405                              BEQ     30$             ; Branch if not host                   ;[TD001]
000166  012701  000044                      MOV     #H$HLEN,R1      ; Length of host homeblock (old format) ;[TD001]
000172  000402                              BR      30$                                                    ;[TD001]
000174  012701  000104              20$:    MOV     #H$GLEN,R1      ; Length of extended homeblock         ;[TD001]
000200                              30$:    CALL    $DEA16          ; Deallocate PSI home block            ;[TD001]
000204  005067  000000G                     CLR     $PFLAG          ; Clear the system type flags          ;[TD001]
000210  005067  000000G                     CLR     $PSIPT          ; Clear PSI home block address         ;[TD001]
000214                              100$:   RETURN                                                         ;[TD001]
```

```
                                                                          C  4
VUNPSI - VNP UNLOAD ROUTINES FO MACRO V05.03b Wednesday 17-Jul-85 12:14  Page 17-2
Symbol table

T$RFRM  000007      T.MXSZ  000050      T3.CLI= 001000      XT$DUC= 000010      X$RTRY  000002
T$RLIN  000004      T.NAM   000006      T3.GFL= 000010      XT$PVC= 000001      X$RXI   000104
T$RSTS  000005      T.OFF   000066      T3.MCR= 004000      XX29    001476R     X$RXQ   000072
T$RTYP  000001      T.PCB   000046      T3.NET= 000020      X$ABQ   000076      X$SS    000004
T$SBUF  000001      T.PRI   000002      T3.NSD= 000200      X$ALF   000055      X$ST    000005
T$SDCT  000030      T.RCVL  000012      T3.PMD= 040000      X$AUC   000056      X$TCLZ  000030
T$SDDM  000026      T.RRFL  000072      T3.PRV= 010000      X$CLEN= 000025      X$TIMC  000001
T$SDEV  000002      T.SAST  000054      T3.REM= 020000      X$CTIM  000060      X$TIMR  000000
T$SDUN  000031      T.SRCT  000071      T3.ROV= 000040      X$DIAG  000003      X$TXI   000102
T$SFNC  000000      T.STAT  000032      T3.RST= 000400      X$DTE   000024      X$TXQ   000066
T$SFNM  000007      T.ST2   000034      T3.SLV= 002000      X$FLG   000014      X$TYP   000015
T$SFTP  000020      T.ST3   000036      T3.SWS= 000002      X$GLEN  000106      X$USR   000022
T$SGRP  000006      T.TCBL  000030      V$$CTR= 001000      X$LCN   000026      X$WAQ   000062
T$SLEN  000032      T.TIO   000057      XA$NCO= 000020      X$LEN   000106      X$WSZ   000020
T$SMEM  000005      T.TKSZ  000060      XA$RR = 000200      X$MOWN  000023      X$$DBT= 000000
T$SNOD= 000023      T.UCB   000026      XA$UCL= 000040      X$NLRE  000052      $BFR  = ****** GX
T$STID  000023      T2.ABO= 000100      XA$URE= 000100      X$NNRE  000054      $CEACC= ****** GX
T$STIU  000025      T2.AST= 100000      XF$NCO= 000020      X$NPL   000006      $DEA16= ****** GX
T$SUNT  000004      T2.CAF= 000400      XF$OFF= 000200      X$NPR   000007      $DEPOL= ****** GX
T$$KMG= 000000      T2.CHK= 020000      XF$ORI= 000002      X$NRBY  000032      $GETAD= ****** GX
T$$MIN= 00000C      T2.CKD= 010000      XF$OTI= 000001      X$NRPK  000042      $GETRV= ****** GX
T.ACTL  000052      T2.DST= 040000      XF$TIL= 000010      X$NRRE  000053      $PFLAG= ****** GX
T.ASTL  000016      T2.FXD= 002000      XF$TIR= 000004      X$NTBY  000036      $PSIPT= ****** GX
T.ATT   000062      T2.HLT= 000200      XLDTE   000772R     X$NTPK  000046      $PUTRC= ****** GX
T.CPCB  000004      T2.REX= 001000      XPORT   001260R     X$PKSZ  000016      $RADDR= ****** GX
T.DPRI  000040      T2.SEF= 004000      XPSV    001720R     X$PR    000011      $RBLCK= ****** GX
T.EFLG  000022      T2.SPN= 000004      XTRA    001554R     X$PRT   000021      $SAVAL= ****** GX
T.IOC   000003      T2.STP= 000020      XT$CHN= 000100      X$PS    000010      $XPSI   000000RG
T.LBN   000041      T2.WFR= 000001      XT$DGR= 000002      X$RPR   000013      $$$   = 000062
T.LDV   000044      T3.ACP= 100000      XT$FAS= 000200      X$RPS   000012      $$$R  = 000010
T.LNK   000000      T3.CAL= 000100      XT$INC= 000004

. ABS.  000134    000   (RW,I,GBL,ABS,OVR)
        002166    001   (RW,I,LCL,REL,CON)
DATA    000004    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 72
Work  file writes: 74
Size of work file: 21046 Words  ( 83 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:51.53
DB2:VUNPSI.T2,[132,134]VUNPSI/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VUNPSI
```

```
                                                           D  4
VUNPSI     CREATED BY  MACRO  ON 17-JUL-85 AT 12:15     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF    04.00
```

```
                                                                C  5
VVCT - VNP VECTOR SETUP ROUTINE MACRO V05.03b Saturday 29-Jun-85 03:12  Page 9-1
Symbol table

A$$CHK= 000000          G$$WRD= 000000          L.MPF = ****** GX       Q$$OPT= 000010          $DDMPV= ****** GX
A$$CPS= 000000          I$$RAR= 000000          L.NMST= ****** GX       R$$DER= 000000          $DLCPV= ****** GX
A$$PRI= 000000          I$$RDN= 000000          L.NSTA= ****** GX       R$$K11= 000001          $FLAGS= ****** GX
A$$TRP= 000000          KRBB    000040R     002 L.CWNR= ****** GX       R$$SND= 000000          $FLAG1= ****** GX
C$$CKP= 000000          K$$CNT= 177546          MANINI  000512R         R$$11M= 000000          $ICBAD= ****** GX
C$$ORE= 000400          K$$CSR= 177546          M$$CRB= 000124          SYSFDB= ****** GX       $LLCTA= ****** GX
C$$RSH= 177564          K$$LDC= 000000          M$$CRX= 000000          S$$WRG= 000000          $MXADD= ****** GX
D$$BUG= 177514          K$$TPS= 000074          M$$FCS= 000000          S$$YSZ= 007600          $MXLEN= ****** GX
D$$ISK= 00   0          LD$LP = 000000          M$$MGE= 000000          S.LEN = ****** GX       $NVECT= ****** GX
D$$L11= 0C   1          LF.RDY= ****** GX       M$$NET- 000000          S.NMST= ****** GX       $OFF  = ****** GX
D$$YNC= 000000          LN.OFF= ****** GX       M$$OVR= 000000          S.OWNR= ****** GX       $PDVA = ****** GX
D$$YNM= 000000          L$$ASG= 000000          N$$ACC= 000001          T$$KMG= 000000          $PUTRC= ****** GX
E$$XPR= 000000          L$$DRV= 000000          N$$BUF= 000001          T$$MIN= 000000          $RADDR= ****** GX
FL.CHA= ****** GX       L$$P11= 000001          N$$LDV= 000001          VCTBUF  000000R     002 $SADDR= ****** GX
FL.DDM= ****** GX       L$$11R= 000000          N$$MCP= 000001          V$$CTR= 001000          $SLTA = ****** GX
FL.DLC= ****** GX       L.CTL = ****** GX       N$$MLL= 000001          X$$DBT= 000000          $SLTMA= ****** GX
FL.MUX= ****** GX       L.DDM = ****** GX       N$$MOV= 000010          ZF.PSE= ****** GX       $SLTNM= ****** GX
F$$LVL= 000001          L.DDS = ****** GX       N$$NCT= 000001          Z.DAT = ****** GX       $TMLTA  000142RG
F.NRBD= ****** GX       L.DLC = ****** GX       N$$PEM= 000001          Z.DSP = ****** GX       $TMRDY  000434RG
F.RSIZ= ****** GX       L.DLM = ****** GX       PR7   = ****** GX       Z.FLG = ****** GX       $TMVCT  000000RG
G$$TPP= 000000          L.DLS = ****** GX       P$$P45= 000000          Z.PCB = ****** GX       $VECT1= ****** GX
G$$TSS= 000000          L.FLG = ****** GX       P$$WRD= 000000          $BFR  = ****** GX       $$VECT= ****** GX
G$$TTK= 000000

. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        000642    001   (RW,I,LCL,REL,CON)
DATA    000120    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10694 Words  ( 42 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:11.18
SY:VVCT.V2,[132,134]VVCT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VVCT
```

```
                                                        D  5
VVCT        CREATED BY  MACRO  ON 29-JUN-85 AT 03:12       PAGE 1

SYMBOL CROSS REFERENCE                                     CREF    04.00
```

Table of contents

```
    6-  120    DELIBV - DEALLOCATE LIBRARY VECTOR BLOCK
```

C 7

```
VXPR       CREATED BY  MACRO  ON 29-JUN-85 AT 03:12     PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE              REFERENCES

FE.PLA  = 000020            5-89
F.NRBD  = ******   GX      *5-97       *5-99
F.RSIZ  = ******   GX      *5-74       *5-78
F.URBD  = ******   GX      *5-77       *5-79
PS.SYS  = 000040            5-83
P.BUSY    000024            5-86       *5-86        5-87        *5-87
P.MAIN    000012            5-75
P.STAT    000030            5-83
P.SUB     000010            5-76        5-107       5-109       *5-113
R$$MPL  = ******            5-81
SYSFDB  = ******   GX      *5-74       *5-77       *5-78        *5-79       *5-97       *5-99
$BFR    = ******   GX       5-75        5-76       *5-76         5-77        5-79        5-83        5-86       *5-86        5-87
                           *5-87       *5-93        5-97         5-99        5-107       5-109       5-113      *5-113
$DEA16  = ******   GX       5-116
$FMASK  = ******   GX       5-89
$GETRV  = ******   GX       5-74        5-78        5-106
$PUTRC  = ******   GX       5-96        5-98        5-114
$RADDR  = ******   GX      *5-74       *5-78       *5-98        *5-106
$XPAR     000000 RG        #5-73
.PLGTH  = ******   GX       5-74        5-78        5-115
.PFRO   = ******   GX       5-92
```

D 7

```
VXPR       CREATED BY  MACRO  ON 29-JUN-85 AT 03:12     PAGE 2
```

```
NTICEX  MACRO V05.03b Saturday 29-Jun-85 03:25  Page 8-2
Symbol table

N$$MCP= 000001          PUMR   = ****** GX      S$$YSZ= 007600        ZF.DVP= 100000         Z.LEN = 000016
N$$MLL= 000001          P$$P45= 000000          S.COST  000001        ZF.INI= 040000         Z.LLN   000006
N$$MOV= 000010          P$$WRD= 000000          S.FLG   000000        ZF.KMX= 000020         Z.MAP   000020
N$$NCT= 000001          Q$$OPT= 000010          S.LEN   000004        ZF.LLC= 000004         Z.NAM   000004
N$$PEM= 000001          R$$DER= 000000          S.NMST  000002        ZF.LMC= 000100         Z.PCB   000012
PDVID  = ****** GX      R$$K11= 000001          S.OWNR  000003        ZF.MAN= 020000         Z.SCH   000007
PDVNM  = ****** GX      R$$SND= 000000          TPS    = 177564       ZF.MFL= 000010         $ERR10  000000R      002
PDVTA  = ****** GX      R$$11M= 000000          T$$KMG= 000000        ZF.MTM= 000400         $NICEX  000000RG
PIRQ   = 177772         SF.ACT= 000200          T$$MIN= 000000        ZF.MUX= 000040         $NICSR= ****** GX
PMODE  = 030000         SF.ENA= 000100          UBMPR  = 170200       ZF.PSE= 002000         $NIERR= * **** GX
PR0    = 000000         SF.LPB= 000004          UISAR0= 177640        ZF.SLI= 010000         $NIFLG= ****** GX
PR1    = 000040         SF.MFL= 000040          UISAR1= 177642        ZF.TIM= 000200         $NIMFL= ****** GX
PR2    = 000100         SF.PAC= 000020          UMRPT  = ****** GX    ZF.X3P= 000000         $NINTL= ****** GX
PR3    = 000140         SF.REA= 000010          V$$CTR= 001000        ZS.ASN= 100000         $NITIM= ****** GX
PR4    = 000200         SF.SER= 000001          X$$DBT= 000000        ZS.BSY= 140000         $NIUMR= ****** GX
PR5    = 000240         SF.SVC= 000002          ZF.COU= 001000        Z.AVL    000014        $NTLFC= ****** GX
PR6    = 000300         SF.UNL= 000040          ZF.DDM= 000001        Z.DAT    000016        .BASEB= 140000
PR7    = 000340         SWR    = 177570         ZF.DIA= 004000        Z.DSP    000000        .CXCSR= ****** GX
PS     = 177776         S$$WRG= 000000          ZF.DLC= 000002        Z.FLG    000010        .$$$$.= 000034

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000312    001   (RW,I,LCL,REL,CON)
DATA    000032    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 52
Work  file writes: 60
Size of work file: 21022 Words  ( 83 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:20.08
SY:NTICEX.V2,[132,134]NTICEX/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTICEX
```

```
NTICEX      CREATED BY  MACRO  ON 29-JUN-85 AT 03:25      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF   04.00
```

```
                                                                      C  9
NTICSR      CREATED BY  MACRO  ON 29-JUN-85 AT 03:25        PAGE 2
SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL   VALUE              REFERENCES

L.DLS      000010           #5-62
L.FLG      000000           #5-62
L.KRBA     000016           #5-62
L.LEN    = 000022           #5-62
L.MPF      000022           #5-62
L.NMST     000020           #5-62
L.NSTA     000014           #5-62
L.OWNR     000021           #5-62
L.UNT      000013           #5-62           6-177
MPAR     = 172100           #5-61
MPCSR    = 177746           #5-61
M$$MGE   = 000000            5-61
PDVTA    = ******    GX      6-163
PIRQ     = 177772           #5-61
PMODE    = 030000           #5-61
PR0      = 000000           #5-61
PR1      = 000040           #5-61
PR2      = 000100           #5-61
PR3      = 000140           #5-61
PR4      = 000200           #5-61
PR5      = 000240           #5-61
PR6      = 000300           #5-61
PR7      = 000340           #5-61
PS       = 177776           #5-61
R$$MPL   = ******            5-61           5-61          5-61
R$$11D   = ******            5-61           5-63
R$$11M   = 000000            5-63
R$$11S   = ******            5-63           6-169
SF.ACT   = 000200           #5-62
SF.ENA   = 000100           #5-62
SF.LPB   = 000004           #5-62
SF.MFL   = 000040           #5-62
SF.PAC   = 000020           #5-62
SF.REA   = 000010           #5-62
SF.SER   = 000001           #5-62
SF.SVC   = 000002           #5-62
SF.UNL   = 000040           #5-62
SWR      = 177570           #5-61
S$$BAS   = ******            5-88           5-88          6-159          6-183
S.COST     000001           #5-62
S.FLG      000000           #5-62
S.LEN      000004           #5-62
S.NMST     000002           #5-62
S.OWNR     000003           #5-62
S.VTTA   = 000002            5-74           5-75
TPS      = 177564           #5-61
UBMPR    = 170200           #5-61
UISAR0   = 177640           #5-61
UISAR1   = 177642           #5-61
X$$MCR   = ******            5-63           5-63
ZF.COU   = 001000           #5-63
ZF.DDM   = 000001           #5-63
```

```
                                                                      D  9
NTICSR      CREATED BY  MACRO  ON 29-JUN-85 AT 03:25        PAGE 3
SYMBOL CROSS REFERENCE                                      CREF    04.00
```

```
                                                 .SBTTL  FNDPLI - GET PLI'S PDV INDEX
                                         ;+
                                         ;
                                         ; FNDPLI - GET PDV INDEX FOR PLI PROCESS
                                         ;
                                         ; INPUTS:
                                         ;       NONE
                                         ;
                                         ; OUTPUTS:
                                         ;       R1 - PDV INDEX FOR PLI
                                         ;
                                         ;-

                                                 .IF NDF R$$11S

000610                                   FNDPLI: SWSTK$  20$                     ;; ENTER SYSTEM STATE
000614  012702  062751                           MOV     #^RPLI,R2               ;; GET RAD50 PROCESS NAME
000620                                           CALL    @PDVID                  ;; LOOK FOR IT
000624  103006                                   BCC     10$                     ;; BR IF FOUND IT

000626                                           RETC    R0                      ;; RETURN CARRY SET
000640  000402                                   BR      20$                     ;; AND EXIT

000642  010266  000004                   10$:    MOV     R2,4(SP)                ;; STORE PDV INDEX IN USER R1

000646                                   20$:    RETURN                          ;;

                                                 .ENDC

        000001                                   .END
```

```
                                    .SBTTL  DATA

                            ;
                            ; IF MEMORY MANAGEMENT IS IN EFFECT, ALWAYS ASSEMBLE CODE AS IF
                            ; IT WERE FOR THE 11/70
                            ;
            000000          .IIF    DF,M$$MGE,      M$$EXT=0        ;ALWAYS ASSEMBLE FOR THE 11/70

                            ;
                            ; LIBRARY MACROS
                            ;
                                    .MCALL  SLTDF$,PDVDF$,RETC,WSIG$S,CCBDF$,SAVRG,RESRG,XPDDB$,ADJDF$
                                    .MCALL  BIAS,MAP,SAVMAP,RESMAP,CTRDF$,SVCDF$,PHBDF$,PLBDF$
                                    .MCALL  MRKT$S,WTSE$S,EMSG$,DHBDF$,ECDDB$,CALLR

000000                              DHBDF$                          ;DEFINE DECNET HOME BLOCK OFFSETS
000000                              ECDDB$                          ;DEFINE ECL DATA BASE OFFSETS
000000                              CCBDF$                          ;DEFINE CCB OFFSETS AND CONSTANTS
000000                              CTRDF$                          ;DEFINE COUNTER BLOCK OFFSETS
000000                              XPDDB$                          ;DEFINE TRANSPORTS DDB OFFSETS
000000                              SLTDF$                          ;DEFINE SLT OFFSETS AND CONSTANTS
000000                              SVCDF$                          ;DEFINE SVC OFFSETS
000000                              PHBDF$                          ;DEFINE PSI HOME BLOCK OFFSETS
000000                              PDVDF$                          ;DEFINE PDV OFFSETS AND CONSTANTS
000000                              PLBDF$                          ;DEFINE PLB OFFSETS
000000                              ADJDF$                          ;DEFINE ADJACENCY BLOCK OFFSETS

                            ;
                            ; CONSTANT DEFINITIONS
                            ;
            000004                  STRT = 4                        ;START STATUS FLAG FOR CCB
            000001                  TMEFN = 1                       ;TEMPORARY EVENT FLAG

                            ;
                            ; STATION ADDRESS
                            ;
000000                      STAADD: .BLKW   1                       ;ADDRESS OF STATION TABLE FROM SLT
000002                      STANUM: .BLKB   1                       ;STATION NUMBER
000003                      CTLNUM: .BLKB   1                       ;CONTROLLER NUMBER
000004      000000          ACTCIR: .WORD   0                       ;XPT CIRCUIT ACTIVE, ENDNODE ONLY


000006                              .PSECT
```

```
                                                                                C 12
NTIMFE      CREATED BY  MACRO  ON 24-JUL-85 AT 15:09      PAGE 4

SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

ZF.COU   = 001000           #5-77
ZF.DDM   = 000001           #5-77
ZF.DIA   = 004000           #5-77
ZF.DLC   = 000002           #5-77
ZF.DVP   = 100000           #5-77
ZF.INI   = 040000           #5-77
ZF.KMX   = 000020           #5-77
ZF.LLC   = 000004           #5-77
ZF.LMC   = 000100           #5-77
ZF.MAN   = 020000           #5-77
ZF.MFL   = 000010           #5-77
ZF.MTM   = 000400           #5-77
ZF.MUX   = 000040           #5-77
ZF.PSE   = 002000           #5-77      7-280
ZF.SLI   = 010000           #5-77
ZF.TIM   = 000200           #5-77
ZF.X3P   = 000000           #5-77
ZS.ASN   = 100000           #5-77
ZS.BSY   = 140000           #5-77
Z.AVL      000014           #5-77
Z.DAT      000016           #5-77      7-285      8-343
Z.DSP      000000           #5-77      5-77
Z.FLG      000010           #5-77      7-280
Z.LEN    = 000016           #5-77
Z.LLN      000006           #5-77
Z.MAP      000020           #5-77      8-336
Z.NAM      000004           #5-77      6-179      6-223
Z.PCB      000012           #5-77
Z.SCH      000007           #5-77
$DBECL   = ******    GX      6-182     6-226
$ERR12   = ******    GX      6-193     6-237
$HEADR   = ******    GX      7-299     8-333
$NIMFE     000006 RG        #6-111
$SAVAL   = ******    GX      6-117
$$$ARG   = 000004           #6-113     6-113      6-113      6-113      6-113      6-113      6-113      6-113      6-113
                             6-113     6-113
```

```
                                                                                D 12
NTIMFE      CREATED BY  MACRO  ON 24-JUL-85 AT 15:09      PAGE 5
MACRO CROSS REFERENCE                                     CREF   04.00
```

```
                                                                  C 13
NTIMFL      CREATED BY  MACRO  ON 29-JUN-85 AT 03:27       PAGE 2
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL   VALUE              REFERENCES

LR.TCB      000002          #5-67
LR.UNT      000005          #5-67       *8-149       8-160
LS.AST   =  100000          #5-67
LS.CEX   =  000001          #5-67
LS.CIR   =  000010          #5-67        9-195
LS.CXO   =  010000          #5-67
LS.CX5   =  000400          #5-67
LS.DLM   =  004000          #5-67        9-195
LS.ECH   =  001000          #5-67
LS.FDX   =  004000          #5-67
LS.HDX   =  010000          #5-67
LS.LIN   =  000003          #5-67        8-144
LS.LMC   =  000007          #5-67
LS.NTI   =  004000          #5-67
LS.ON    =  000011          #5-67
LS.OPT   =  000400          #5-67
LS.PRO   =  000002          #5-67        7-117
LS.PWF   =  040000          #5-67
LS.TOP   =  002000          #5-67        7-120        8-147        9-198
LS.UNF   =  020000          #5-67
LS.X25   =  000012          #5-67
LS.11D   =  004000          #5-67
LX.CEX   =  000004          #5-67
LX.LIN   =  000006          #5-67
LX.PRO   =  000005          #5-67
L.COST      000015          #5-65
L.CTL       000012          #5-65        8-148        9-200
L.CVA       177776          #5-65
L.DDM       000002          #5-65        8-136
L.DDS       000004          #5-65
L.DLC       000003          #5-65
L.DLM       000006          #5-65
L.DLS       000010          #5-65
L.FLG       000000          #5-65        8-134
L.KRBA      000016          #5-65
L.LEN    =  000022          #5-65
L.MPF       000022          #5-65        9-191
L.NMST      000020          #5-65
L.NSTA      000014          #5-65        9-188
L.OWNR      000021          #5-65
L.UNT       000013          #5-65        8-149
MPAR     =  172100          #5-66
MPCSR    =  177746          #5-66
M$$MGE   =  000000           5-66
PDVFLG      000000 R        #6-74       *8-151       8-161
PDVNM    =  ******   GX      7-112
PDVTA    =  ******   GX      7-111        8-137
PIRQ     =  177772          #5-66
PMODE    =  030000          #5-66
PR0      =  000000          #5-66
PR1      =  000040          #5-66
PR2      =  000100          #5-66
```

```
                                                                  D 13
NTIMFL      CREATED BY  MACRO  ON 29-JUN-85 AT 03:27       PAGE 3
SYMBOL CROSS REFERENCE                                     CREF    04.00
```

```
    316                                                                 ; THE ROUTINE NTIRDX AND FROM ABOVE
    317                                 MOV     #$NIRTN,R0              ;POINT AT SENDING TASK'S NAME
    318                                 MOV     (R0)+,NTISEN+S.DATN     ;SAVE SENDER'S NAME IN
    319                                 MOV     (R0)+,NTISEN+S.DATN+2   ;  RETURN DIRECTIVE
    320                                 MOV     (R0)+,$NIFNC            ;SAVE FUNCTION CODE
    321                                 MOV     (R0)+,$NIFNC+2          ; SAVE TWO WORDS OF SENDER'S DATA
    322                                 MOV     (R0),$NIFNC+4           ; (USED FOR POWERFAIL RECOVERY)
    323
    324                                 CLR     $NIAST                  ;ASSUME CALLER NOT NOTIFIED VIA AST
    325                                 BIT     #LS.AST,$NIFNC          ;NOTIFY CALLER VIA AST?
    326                                 BEQ     20$                     ;BR IF NO
    327                                 MOV     LR.AST+$NIRDB,$NIAST    ;STORE ADDRESS OF NOTIFICATION AST
    328                                 MOV     LR.PRM+$NIRDB,$NIPRM    ;STORE NOTIFICATION AST PARAMETER
    329                         20$:
    330                                 .ENDC ;DF S$$BAS
    331                                                                 ;FALL THROUGH
    332                                                                 ;INTO MAIN LOOP .........
```

```
   829                                      .ENDC
```

```
                                                                              C 16
**FILE**ID**NTITIM




NN      NN   TTTTTTTTTT     IIIIII     TTTTTTTTTT     IIIIII     MM      MM
NN      NN   TTTTTTTTTT     IIIIII     TTTTTTTTTT     IIIIII     MM      MM
NN      NN       TT           II           TT           II       MMMM  MMMM
NN      NN       TT           II           TT           II       MMMM  MMMM
NNNN    NN       TT           II           TT           II       MM  MM  MM
NNNN    NN       TT           II           TT           II       MM  MM  MM
NN  NN  NN       TT           II           TT           II       MM      MM
NN  NN  NN       TT           II           TT           II       MM      MM
NN    NNNN       TT           II           TT           II       MM      MM
NN    NNNN       TT           II           TT           II       MM      MM
NN      NN       TT           II           TT           II       MM      MM       ....
NN      NN       TT           II           TT           II       MM      MM       ....
NN      NN       TT         IIIIII         TT         IIIIII     MM      MM       ....
NN      NN       TT         IIIIII         TT         IIIIII     MM      MM       ....




LL             SSSSSSSS    TTTTTTTTTT
LL             SSSSSSSS    TTTTTTTTTT
LL           SS                TT
LL           SS                TT
LL           SS                TT
LL           SS                TT
LL             SSSSSS          TT
LL             SSSSSS          TT
LL                   SS        TT
LL                   SS        TT
LL                   SS        TT
LL                   SS        TT
LLLLLLLLLL   SSSSSSSS          TT
LLLLLLLLLL   SSSSSSSS          TT
```

```
                                                                              D 16
NTITIM  MACRO V05.03b Saturday 29-Jun-85 03:28
Table of contents
```

```
   1578                                                  .SBTTL  FLT16 - FIND LINE TABLE
   1579                                          ;+
   1580                                          ; FLT16 - FIND A 16 BIT ALLOCATION LINE TABLE
   1581                                          ;
   1582                                          ; INPUTS:
   1583                                          ;        R3=LINE TABLE ADDRESS
   1584                                          ;        R4 = PDV INDEX OF PROCESS
   1585                                          ;
   1586                                          ; OUTPUTS:
   1587                                          ;        R0,R1,R2,R3=DESTROYED
   1588                                          ;        UBIAS,UBA=UNLOAD BLOCK ADDRESS
   1589                                          ;        BLOCK=LOCAL COPY OF UNLOAD BLOCK
   1590                                          ;-
   1591
   1592 005242                                   FLT16:
   1593 005242  005067  000010'                  FLT:    CLR     UBA                     ; INDICATE NO UNLOAD BLOCK FOUND YET
   1594 005246  012702  000012'                          MOV     #BLOCK,R2               ; USE R2 AS BASE REGISTER
   1595 005252  012767  000100  000000G                  MOV     #64.,$RSIZE             ; SIZE OF UNLOAD BLOCK IN BYTES
   1596 005260                                           SWTCHI  R2                      ; USE BLOCK AS INPUT BUFFER
   1597 005264  005077  000000G                          CLR     @$ADRZ                  ; SET UP ADDRESS
   1598 005270  016777  000000C 000000G                  MOV     .CXUNL+$NTLHB,@$ADRX ; OF UNLOAD BLOCK
   1599 005276  001461                                   BEQ     100$                    ; NO UNLOAD BLOCKS
   1600
   1601 005300                                   10$:    GETAD                           ; READ IN IMAGE OF UNLOAD BLOCK
   1602 005304  005722                                   TST     (R2)+                   ; SKIP POINTER TO NEXT UNLOAD BLOCK
   1603 005306  005722                           20$:    TST     (R2)+                   ; POINT PAST THE PDV INDEX
   1604 005310  126204  177776                           CMPB    -2(R2),R4               ; IS THIS THE RIGHT PDV ?
   1605 005314  001013                                   BNE     30$                     ; IF NE, NO
   1606 005316  022203                                   CMP     (R2)+,R3                ; IS THIS OUR LINE TABLE?
   1607 005320  001440                                   BEQ     80$                     ; IF EQ, YES
   1608 005322  101011                                   BHI     40$                     ; IF HI, NO
   1609 005324  011246                                   MOV     (R2),-(SP)              ; EXTRACT SIZE FIELD
   1610 005326  042716  000003                           BIC     #3,(SP)                 ; ...
   1611 005332  066216  177776                           ADD     -2(R2),(SP)             ; ADD TABLE ENTRY
   1612 005336  022603                                   CMP     (SP)+,R3                ; IS THIS OUR LINE TABLE?
   1613 005340  101030                                   BHI     80$                     ; IF HI, YES
   1614 005342  000401                                   BR      40$                     ; ELSE, SKIP ANY SUB-ALLOCATIONS
   1615 005344  005722                           30$:    TST     (R2)+                   ; SKIP THE LINE TABLE ADDRESS
   1616 005346  032722  000001                   40$:    BIT     #1,(R2)+                ; ANY SUB-ALLOCATIONS THIS ENTRY?
   1617 005352  001410                                   BEQ     50$                     ; IF EQ, NO
   1618 005354  005046                                   CLR     -(SP)                   ; GET NUMBER OF ALLOCATIONS
   1619 005356  152216                                   BISB    (R2)+,(SP)              ; ...
   1620 005360  005046                                   CLR     -(SP)                   ; AND NUMBER OF LIBRARIES
   1621 005362  152216                                   BISB    (R2)+,(SP)              ; ...
   1622 005364  062616                                   ADD     (SP)+,(SP)              ; ADD THEM TOGETHER
   1623 005366  006316                                   ASL     (SP)                    ; MULTIPLY BY 4
   1624 005370  006316                                   ASL     (SP)                    ; ...
   1625 005372  062602                                   ADD     (SP)+,R2                ; GIVING NEXT TABLE ENTRY
   1626 005374  022702  000112'                  50$:    CMP     #BLOCK+100,R2           ; IS IT PHYSICAL END OF BLOCK?
   1627 005400  001402                                   BEQ     60$                     ; IF EQ, YES
   1628 005402  005712                                   TST     (R2)                    ; IS IT LOGICAL END OF BLOCK?
   1629 005404  001340                                   BNE     20$                     ; IF NE, NO - CHECK NEXT ENTRY
   1630 005406  012702  000012'                  60$:    MOV     #BLOCK,R2               ; RESET R2 TO START OF BLOCK
   1631                                          ;*;     CLR     @$ADRZ                  ; SET NEW ADDRESS; THIS CAUSES BUG;
   1632 005412  011277  000000G                          MOV     (R2),@$ADRX             ; SET NEW ADDRESS
   1633 005416  001330                                   BNE     10$                     ; READ IN BLOCK
   1634 005420  000410                                   BR      100$                    ; DONE
```

```
                                                               D 2
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31      PAGE 4
SYMBOL  CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE            REFERENCES

T$LLDL     000012           #5-85
T$LLDO     000012           #5-85
T$LLDS     000012           #5-85
T$LLEN     000046           #5-85         10-471
T$LOPR     000002           #5-85
T$LTCL     000024           #5-85
T$LTIM     000026           #5-85
T$LTPR     000014           #5-85
T$LTPS     000020           #5-85
T$NAPL     000004           #5-85
T$NFE      000000           #5-85
T$NLEN     000010           #5-85
T$NNUL     000002           #5-85
T$NOPL     000006           #5-85
T$NRNI     000042           #5-85
T$NRPL     000005           #5-85
T$NRUL     000007           #5-85
T$NVR      000001           #5-85
T$RPRI     000040           #5-85
T$SVC      000034           #5-85
T$T5       000030           #5-85
T$T6       000032           #5-85
T.NAM      000006            25-1857       25-1859
T.RCVL     000012            21-1265      *21-1267       21-1269      *21-1270
T.TCBL     000030           *25-1851       25-1853
UBA        000010 R         #6-133         17-995       *22-1593      *22-1637       23-1655
UBIAS      000006 R         #6-132        *22-1638       23-1731       23-1734       23-1749       23-1774
UNITS      000004 R         #6-127        *15-859       *15-873        15-878
UPTR       000112 R         #6-143        *23-1658       23-1725       23-1765
VECT       002402 R          14-747       #17-953
VF.LIB   = 000001             7-210
VF.SS    = ******   GX        8-314
V.FLG      000002             7-210
V.LEN      000012             7-208         7-233
V.PDV      000004             7-215
XALITB     004660 R          21-1345      #21-1508
XDCB       004456 R          21-1308      #21-1465
XDLC       000360 RG         8-315          8-317        8-345         8-348        #8-351         12-584        14-777
XFLTTB     005150 R          21-1348      #21-1561
XKRB       002162 R          12-563        14-749       #14-801
XLIN       001614 R         #14-651        21-1294
XLT        005456 R          14-751       #23-1655
XLT16      005452 R          8-383         12-576        14-769       #23-1654
XOBJTB     004576 R          21-1340      #21-1490
XPRO       000152 R          8-265        #8-268         12-589        13-618        14-759        21-1309
XREMTB     005022 R          21-1346      #21-1537
ZDDS       003354 R          14-752       #19-1174
ZF.LLC   = ******   GX       8-364
ZF.LMC   = ******   GX       8-316
ZF.MFL   = ******   GX       13-619        14-757        14-775
ZF.PSE   = ******   GX       14-655
Z.AVL    = ******   GX       24-1815
```

```
                                                               E 2
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31      PAGE 5
SYMBOL  CROSS REFERENCE                                   CREF    04.00
```

```
                                                 .SBTTL  DEAPVC - Deallocate PVC name blocks                 ;[TD001]
                                        ;+                                                                   ;[TD001]
                                        ; *** DEAPVC - Deallocate PVC name blocks                            ;[TD001]
                                        ;                                                                    ;[TD001]
                                        ; INPUTS                                                             ;[TD001]
                                        ;        $PFLAG - PSI characteristics word                           ;[TD001]
                                        ;                                                                    ;[TD001]
                                        ; OUTPUTS                                                            ;[TD001]
                                        ;        PVC name blocks deallocated                                 ;[TD001]
                                        ;                                                                    ;[TD001]
                                        ;        All registers preserved                                     ;[TD001]
                                        ;-                                                                   ;[TD001]
 000216                                 DEAPVC: CALL    $SAVAL            ; Save all registers               ;[TD001]
 000222  016703  000000G                        MOV     $PSIPT,R3         ; Get address of listhead          ;[TD001]
 000226  062703  000006                         ADD     #H$PVC,R3                                            ;[TD001]
 000232  032767  000000G 000000G                BIT     #PF.XDF,$PFLAG    ; Check for extended data format   ;[TD001]
 000240  001005                                 BNE     10$               ; Branch if extended               ;[TD001]
                                        ;                                                                    ;[TD001]
                                        ; Deallocate old-format PVC name blocks                              ;[TD001]
                                        ;                                                                    ;[TD001]
 000242  012701  000016                         MOV     #C$LEN,R1         ; Length of old-format block       ;[TD001]
 000246                                         CALL    DEALST            ; Deallocate                       ;[TD001]
 000252  000447                                 BR      60$                                                  ;[TD001]
                                        ;                                                                    ;[TD001]
                                        ; Deallocate extended PVC name blocks                                ;[TD001]
                                        ;                                                                    ;[TD001]
 000254                                 10$:    SAVMAP                    ; Save current mapping             ;[TD001]
 000260                                         GETRV   R3,#2             ; Read in listhead                 ;[TD001]
 000276  016746  000000G                        MOV     $BFR,-(SP)        ; Save address                     ;[TD001]
 000302  005067  000000G                        CLR     $BFR              ; Zero buffer and rewrite listhead ;[TD001]
 000306                                         PUTRC                                                        ;[TD001]
                                                                                                             ;[TD001]
 000312  012600                         20$:    MOV     (SP)+,R0          ; Restore address of block         ;[TD001]
 000314  001424                                 BEQ     50$               ; Branch if end of chain           ;[TD001]
 000316                                         CEACC$   0,R2             ; Map to block                     ;[TD001]
 000326                                         GETAD   R2,#C$LEN         ; Read fixed part of block         ;[TD001]
 000344  016746  000000G                        MOV     $BFR,-(SP)        ; Save link address                ;[TD001]
 000350  116701  000002G                        MOVB    $BFR+C$NML,R1     ; Get length of variable part      ;[TD001]
 000354  062701  000016                         ADD     #C$LEN,R1         ; Add length of fixed part         ;[TD001]
 000360                                         CALL    $DEPOL            ; Deallocate block                 ;[TD001]
 000364  000752                                 BR      20$                                                  ;[TD001]
                                                                                                             ;[TD001]
 000366                                 50$:    RESMAP                    ; Restore mapping                  ;[TD001]
 000372                                 60$:    RETURN                                                       ;[TD001]
```

```
                                                      D  4
VUNPSI     CREATED BY  MACRO  ON 17-JUL-85 AT 12:15      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE            REFERENCES

C$LEN     000016          9-178     9-193     9-196
C$NML     000002          9-195
DEACUG    000374 R        8-120    #10-216
DEADST    000552 R        8-128     8-130    #11-261
DEALST    002046 R        8-123     8-126     8-135     9-179     10-225    #17-537
DEAPVC    000216 R        8-119    #9-170
D$ACL     000027         11-282
D$DATL    000030         11-294
D$DSL     000022         11-279
D$DTEL    000031         11-297
D$FLEN    000032         11-288    11-290
D$GLEN    000032         11-276    11-278
F.RSIZ  = ******   GX    *9-185    *9-193    *10-231   *10-239   *11-263   *11-276   *11-288   *12-330   *12-335
                         *12-347   *12-365   *13-385   *13-388   *13-395   *14-436   *15-464   *15-469   *16-507
                         *16-514   *17-539   *17-547
F.URBD  = ******   GX    *12-327   *12-331
G$LEN     000016         10-224    10-239    10-242
G$NML     000015         10-241
H$CUG     000010         10-218
H$DST     000012          8-127     8-129
H$D29     000014          8-129     8-133
H$GLEN    000104          8-152
H$GLT     000044         16-506
H$GPT     000046         16-513
H$HLEN    000044          8-150
H$LDTE    000002         12-329
H$LEN     000042          8-145
H$NPT     000022         13-387
H$PTB     000020         13-384    13-387
H$PVC     000006          9-172
H$RDTE    000004          8-121     8-124
H$RNW     000042          8-133
H$SVC     000036          8-124     8-127
H$TRB     000016         15-463
H$X29C    000040         14-435
I$$AS   = ******         12-356    15-475
K$LEN     000006         12-349
LSTHBF    000000 R       #7-101    12-327    12-332    *12-336
L$CHLS    000034         12-345   *12-348
L$CHTB    000030         12-353
L$CTEN    000032         12-355
L$GLEN    000134         12-342
L$LEN     000122         12-335    12-339
N$$VCT  = ******          9-192    10-238    11-270    12-334    13-392    17-546
PF.PSV  = ******   GX    12-340    13-403    16-501
PF.XDF  = ******   GX     8-146     9-173    10-219    11-271
PF.XGA  = ******   GX     8-131     8-148
Q$LEN     000016         14-441
R$LEN     000024          8-122
R$NWL     000104          8-134
R$R0    = 000002         #6-91     13-413
R$R1    = 000004         #6-92
```

```
                                                      E  4
VUNPSI     CREATED BY  MACRO  ON 17-JUL-85 AT 12:15      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF   04.00
```

```
                                                                     D  5
VVCT        CREATED BY  MACRO  ON 29-JUN-85 AT 03:12      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE                REFERENCES

FL.CHA  = ******  GX          6-95        7-205
FL.DDM  = ******  GX          6-93        7-213       7-221
FL.DLC  = ******  GX          7-208       7-223
FL.MUX  = ******  GX          7-228
F.NRBD  = ******  GX         *6-117      *6-129
F.RSIZ  = ******  GX         *6-120
KRBB      000040 R           #5-70
LF.RDY  = ******  GX          8-281
LN.OFF  = ******  GX          9-320       9-330
L.CTL   = ******  GX          7-239       7-239
L.DDM   = ******  GX          7-237       7-237       7-242       7-251
L.DDS   = ******  GX         *7-241      *7-250       8-276
L.DLC   = ******  GX          7-242       7-251
L.DLM   = ******  GX         *7-216
L.DLS   = ******  GX         *7-244      *7-253       8-278
L.FLG   = ******  GX         *8-281
L.MPF   = ******  GX          9-313
L.NMST  = ******  GX         *9-330
L.NSTA  = ******  GX          9-306
L.OWNR  = ******  GX         *9-329
MANINI    000512 R            8-280      #9-295
PR7     = ******  GX          6-113
R$$MPL  = ******              6-105       6-119       6-133
SYSFDB  = ******  GX         *6-117      *6-120      *6-129
S.LEN   = ******  GX          9-321
S.NMST  = ******  GX         *9-320
S.OWNR  = ******  GX         *9-319
VCTBUF    000000 R           #5-69        6-98        6-117
ZF.PSE  = ******  GX          6-91
Z.DAT   = ******  GX         *7-226
Z.DSP   = ******  GX          7-216
Z.FLG   = ******  GX          6-91
Z.PCB   = ******  GX          8-271       8-274
$BFR    = ******  GX          6-129
$DDMPV  = ******  GX          8-273
$DLCPV  = ******  GX          7-212       8-269
$FLAGS  = ******  GX          6-93        6-95        7-205       7-208       7-213       7-221       7-223       7-228
$FLAG1  = ******  GX          6-87        7-202
$ICBAD  = ******  GX          6-111
$LLCTA  = ******  GX          9-305
$MXADD  = ******  GX          7-230       7-232
$MXLEN  = ******  GX          7-245
$NVECT  = ******  GX          6-100
$OFF    = ******  GX          7-220
$PDVA   = ******  GX          6-90        7-215       7-225
$PUTRC  = ******  GX          6-120
$RADDR  = ******  GX         *6-120
$SADDR  = ******  GX          7-210       7-218
$SLTA   = ******  GX          7-204
$SLTMA  = ******  GX          7-235       9-297       9-304
$SLTNM  = ******  GX          7-234       9-296
$TMLTA    000142 RG          #7-201
```

```
                                                                     E  5
VVCT        CREATED BY  MACRO  ON 29-JUN-85 AT 03:12      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF    04.00
```

```
                              .TITLE  VXLB - VNP LIBRARY DEALLOCATION ROUTINES
                              .IDENT  /V05.00/

                        ;
                        ; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                        ; TRANSFERRED.
                        ;
                        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                        ; CORPORATION.
                        ;
                        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                        ;
                        ; MODULE DESCRIPTION:
                        ;
                        ;       VNP - LIBRARY DE-ALLOCATION ROUTINES
                        ;
                        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                        ;
                        ; IDENT HISTORY:
                        ;
                        ; 1.00  27-FEB-78
                        ;       VERSION 2.0 RELEASE
                        ;
                        ; 2.00  14-DEC-79
                        ;       DECNET-11M/S V3.0
                        ;       DECNET-11M-PLUS V1.0
                        ;
                        ; 3.00  16-APR-82
                        ;       DECNET-11M V3.1
                        ;       DECNET-11M-PLUS V1.1
                        ;
                        ; 4.00  07-NOV-83
                        ;       DECNET-11M V4.0
                        ;       DECNET-11M-PLUS V2.0
                        ;
                        ; 5.00  22-JUL-85
                        ;       DECnet-11M/S V
                        ;       DECnet-11M-Plu
                        ;       DECnet-Micro/RSX    )
                        ;
                        ;

                              .MCALL  VECDF$,CEACC$,GETAD

        000000                        VECDF$          ; DEFINE VECTOR BLOCK OFFSETS

```

```
VXPR       CREATED BY  MACRO  ON 29-JUN-85 AT 03:12      PAGE 2
MACRO CROSS REFERENCE                                   CREF   04.00
MACRO NAME        REFERENCES

CALL               5-74      5-78      5-96      5-98      5-106     5-114     5-116
GETRV             #5-69      5-74      5-78      5-106
HWDDF$            #5-69      5-71
PCBDF$            #5-69      5-70
PUTRC             #5-69      5-96      5-98      5-114
RAND              #5-74      5-74     #5-78      5-78     #5-96      5-96     #5-98      5-98     #5-106     5-106
                  #5-114     5-114
RETURN             5-117
SWTCHI            #5-69      5-77      5-79
SWTCHO            #5-69      5-97      5-99
```

```
                                                                D  8
NTICEX     CREATED BY  MACRO  ON 29-JUN-85 AT 03:25        PAGE 1
SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL  VALUE                 REFERENCES

BUFUMP  = 172354              #5-74
CMODE   = 140000              #5-74
FC.PCT  = 000030               5-94
FE.CEX  = ******   GX          6-137
FMASK   = ******   GX          6-137
FNDPDV  = ******   GX          6-148         7-188
FS.INI  = 000000               5-94
INIFUN    000030 R            #5-94          7-205
I$$AS   = ******               5-74          5-75
KISAR0  = 172340              #5-74
KISAR6  = 172354              #5-74
KSAR6   = ******   GX          7-228
LF.ACT  = 100000              #5-73
LF.BRO  = 000400              #5-73
LF.BWT  = 000007              #5-73
LF.ENA  = 002000              #5-73
LF.LPB  = 001000              #5-73
LF.MDC  = 000100              #5-73
LF.MFL  = 004000              #5-73
LF.MTP  = 000020              #5-73
LF.PAC  = 000200              #5-73
LF.RDY  = 040000              #5-73
LF.REA  = 010000              #5-73
LF.SER  = 000040              #5-73
LF.TIM  = 000010              #5-73
LF.UNL  = 020000              #5-73
LF.X2P  = 000000              #5-73
LLCAL   = ******   GX          7-207
LN.CLO  = 000000              #5-73
LN.DUM  = 000005              #5-73
LN.LOA  = 000004              #5-73
LN.LOO  = 000003              #5-73
LN.OAU  = 000003              #5-73
LN.OFF  = 000001              #5-73
LN.ON   = 000000              #5-73
LN.OOP  = 000004              #5-73
LN.OPE  = 000001              #5-73
LN.REF  = 000002              #5-73
LN.SER  = 000002              #5-73
LN.STA  = 000017              #5-73
LN.SUB  = 000360              #5-73
LN.TRI  = 000006              #5-73
LS.CEX  = ******   GX          6-139
LS.NTI  = ******   GX          6-139
LS.TOP  = ******   GX          6-145
LX.CEX  = ******   GX          6-154
L.COST    000015              #5-73
L.CTL     000012              #5-73
L.CVA     177776              #5-73
L.DDM     000002              #5-73
L.DDS     000004              #5-73
L.DLC     000003              #5-73
```

```
                                                                E  8
NTICEX     CREATED BY  MACRO  ON 29-JUN-85 AT 03:25        PAGE 2
SYMBOL CROSS REFERENCE                                      CREF    04.00
```

```
                                                                     D  9
NTICSR      CREATED BY  MACRO  ON 29-JUN-85 AT 03:25        PAGE 3
SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL   VALUE                REFERENCES

ZF.DIA   = 004000             #5-63
ZF.DLC   = 000002             #5-63
ZF.DVP   = 100000             #5-63
ZF.INI   = 040000             #5-63
ZF.KMX   = 000020             #5-63
ZF.LLC   = 000004             #5-63
ZF.LMC   = 000100             #5-63
ZF.MAN   = 020000             #5-63
ZF.MFL   = 000010             #5-63
ZF.MTM   = 000400             #5-63
ZF.MUX   = 000040             #5-63
ZF.PSE   = 002000             #5-63
ZF.SLI   = 010000             #5-63
ZF.TIM   = 000200             #5-63
ZF.X3P   = 000000             #5-63
ZS.ASN   = 100000             #5-63
ZS.BSY   = 140000             #5-63
Z.AVL      000014             #5-63
Z.DAT      000016             #5-63
Z.DSP      000000             #5-63       5-63
Z.FLG      000010             #5-63
Z.LEN    = 000016             #5-63
Z.LLN      000006             #5-63
Z.MAP      000020             #5-63
Z.NAM      000004             #5-63       6-165
Z.PCB      000012             #5-63
Z.SCH      000007             #5-63
$ERR20     000020 RG          #5-88       6-189
$NERDN   = ******   GX         5-88
$NICSR     000000 RG          #6-117
$NIERA   = ******   GX        *6-185     *6-186     *6-188
$NIERR   = ******   GX         5-88
$NINTL   = ******   GX         6-179
$NTLFC   = ******   GX        *6-175     *6-176     *6-177     *6-178
$$$GLB   = ******              5-74       5-75
.BASEB   = 140000             #5-61       6-121      6-127      6-129      6-130
```

```
                                                                     E  9
NTICSR      CREATED BY  MACRO  ON 29-JUN-85 AT 03:25        PAGE 4
```

```
                                                                D 10
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:26  Page 9-1
Symbol table

ALOCB = ****** GX         CM.CIR= 000002         C.FNC   000010         D$MAXV  000070         FC.MAN= 000024
AREAMK= 176000            CM.FMT= 100000         C.LIN   000006         D$MLL   000040         FC.MLD= 000026
ASSIGN  000042R       002 CM.HRD= 000002         C.LNK   000000         D$MNOD  000041         FC.PCT= 000030
AT$ACL= 000100            CM.LIN= 000000         C.MOD   000011         D$NA    000062         FC.PWR= 000022
AT$ADN= 000040            CM.LOO= 000001         C.NSP   000004         D$NBEA  000056         FC.RCE= 000002
AT$AUP= 000020            CM.XLO= 000004         C.PRO   000042         D$NBRA  000054         FC.RCP= 000014
AT$CYC= 000004            CP.DCF= 000040         C.RSV   000002         D$NEND= 000054         FC.TIM= 000010
AT$LV1= 000002            CP.HDL= 000007         C.STA   000007         D$NLN   000030         FC.XCP= 000012
AT$LV2= 000001            CP.PS = 177400         C.STS   000012         D$NN    000060         FC.XME= 000000
AT$NEX= 000200            CP.PSI= 000200         C.URM   177776         D$OUTT  000043         FNDPDV= ****** GX
AT$UP = 000010            CP.XCF= 000100         C.XACP  000004         D$RETF  000050         FNDPLI  000610R
A$CIR   000003            CP.2FR= 000030         C.XID   000035         D$RNN   000002         FS.AST= 000000
A$LEN   000004            CS.ABO= 000100         C.XLEN  000044         D$RTMR  000076         FS.CIB= 002000
A$NID   000000            CS.BRO= 000002         C.XPLI  000040         D$SEG   000036         FS.CRA= 001000
A$TM    000003            CS.BUF= 000200         C.XPT   000034         D$SER   000032         FS.DIS= 013000
A$TMI   000002            CS.CES= 000002         C.XSVC  000042         D$SQRL  000052         FS.DVC= 001000
A$TSZ   000000            CS.CHN= 000010         C.XTC   000037         D$$BUG= 177514         FS.ENB= 012000
A$TYP   000002            CS.CMP= 000200         C.X25   000036         D$$ISK= 000000         FS.EXI= 001000
A$$CHK= 000000            CS.DCR= 000400         DEACB = ****** GX      D$$L11= 000001         FS.GET= 006000
A$$CPS= 000000            CS.DEF= 000004         DECPT = ****** GX      D$$YNC= 000000         FS.HLT= 000000
A$$PRI= 000000            CS.DEV= 000002         DV.CCL= 000002         D$$YNM= 000000         FS.INI= 000000
A$$TRP= 000000            CS.DIS= 000040         DV.COM= 020000         D.DSP   000012         FS.KIL= 000000
A.LULU= 000002            CS.ENA= 000001         DV.DIR= 000010         D.LNK   000000         FS.LCL= 100000
A.LUNA= 000004            CS.ENB= 000020         DV.EXT= 000400         D.MSK   000014         FS.LTM= 001000
A.LUNU= 000006            CS.ERR= 100000         DV.F11= 040000         D.NAM   000004         FS.MNT= 004000
BUFUMP= 172354            CS.FTL= 001000         DV.ISP= 002000         D.PCB   000034         FS.MSN= 014000
CB.CCB= 000002            CS.HCR= 000001         DV.MBC= 000400         D.UCB   000002         FS.REA= 001000
CB.DDM= 000040            CS.HFE= 002000         DV.MNT= 100000         D.UCBL  000010         FS.RET= 000000
CB.DLC= 000020            CS.LST= 040000         DV.MSD= 000100         D.UNIT  000006         FS.REZ= 003000
CB.RDB= 000004            CS.MTL= 004000         DV.OSP= 004000         D.VCAN= 000002         FS.RLB= 002000
CB.SDB= 000010            CS.RNG= 000010         DV.PSE= 010000         D.VDEB= 177776         FS.RNG= 011000
CB.SLI= 000100            CS.ROV= 000004         DV.REC= 000001         D.VINI= 000000         FS.RST= 000000
CB.XLB= 000001            CS.RSN= 010000         DV.SDI= 000020         D.VOUT= 000004         FS.RTN= 001000
CC.LLC= 000200            CS.SHU= 000001         DV.SQD= 000040         D.VPWF= 000006         FS.SET= 005000
CE.ABO= 100362            CS.SID= 000002         DV.SWL= 001000         E$NBR   000014         FS.SFC= 005000
CE.DAO= 100346            CS.STR= 000004         DV.TTY= 000004         E$NBS   000020         FS.SFR= 006000
CE.DIS= 100366            CS.SUC= 000001         DV.UMD= 000200         E$NCR   000034         FS.SFS= 004000
CE.ERR= 100370            CS.TMO= 020000         D$AMXC  000072         E$NCS   000036         FS.SPW= 040000
CE.ILN= 100350            CS.XUR= 000004         D$AMXH  000074         E$NIC   000044         FS.STM= 000000
CE.LTO= 100356            C$LEN   000012         D$ANN   000000         E$NLEN  000050         FS.STP= 002000
CE.MOP= 100372            C$NID   000000         D$BRPR  000102         E$NLLA  000012         FS.STR= 001000
CE.NTE= 100361            C$NXH   000002         D$BRTM  000100         E$NLNK  000000         FS.TRM= 003000
CE.RTE= 100376            C$TIM   000010         D$DELF  000045         E$NML   000040         FS.WLB= 001000
CE.SRC= 100364            C$$CKP= 000000         D$DELW  000046         E$NMR   000024         FS.XKL= 002000
CE.STP= 100352            C$$ORE= 000400         D$END = 000104         E$NMS   000030         FS.XOF= 010000
CE.TME= 100354            C$$RSH= 177564         D$FNB   000034         E$NNOD  000002         FS.XON= 007000
CE.TMO= 100374            C.ADD   0[illegible]034    D$HIOR  000024         E$NRT   000042         FS.ZER= 002000
CE.UNS= 100344            C.BID   0[illegible]003    D$HOST  000022         E$NRTP  000005         F$$LVL= 000001
CF.CHN= 000001            C.BUF   000014         D$INAC  000044         E$NSEG  000010         G$$TPP= 000000
CF.EOM= 000004            C.BUF1  000014         D$INCT  000042         E$NTIM  000046         G$$TSS= 000000
CF.HDR= 000020            C.BUF2  000024         D$IPL   000051         E$NUSE  000004         G$$TTK= 000000
CF.LB = 100000            C.CNT   000020         D$LID   000020         E$STRT  000006         G$$WRD= 000000
CF.LIN= 000002            C.CNT1  000020         D$LNAM  000006         E$$XPR= 000000         HEADR = ****** GX
CF.SOM= 000010            C.CNT2  000030         D$LNUM  000014         FC.CCP= 000020         H.CSP   000000
CF.SYN= 000040            C.FLG   000022         D$LST   000047         FC.CTL= 000006         H.CUIC  000010
CF.TRN= 000100            C.FLG1  000022         D$MAXC  000064         FC.KCP= 000016         H.DSW   000046
CMODE = 140000            C.FLG2  000032         D$MAXH  000066         FC.KIL= 000004         H.DUIC  000012
```

```
                                                                E 10
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:26  Page 9-2
Symbol table
```

```
  98                                          .SBTTL  MARKED FOR ENABLE SCAN
  99
 100                                  ;+
 101                                  ; $NIMFE - PERFORM MARKED-FOR-ENABLE SCAN
 102                                  ;
 103                                  ; INPUTS:
 104                                  ;       ALL PROCESS/LINES MARKED FOR LOAD HAVE BEEN LOADED
 105                                  ;
 106                                  ; OUTPUTS:
 107                                  ;       ALL LINES THAT ARE READY AND OWNED BY SPECIFIED LLC ARE ENABLED
 108                                  ;       ALL REGISTERS ARE PRESERVED
 109                                  ;-
 110
 111 000006                           $NIMFE::
 112                                          .IF NDF S$$BAS
 113 000006                                   MRKT$S  #TMEFN,#1,#2              ; Wait 1 second in case system has
 114 000032                                   WTSE$S  #TMEFN                    ;   been VNP'd and contains a DMV
 115                                          .ENDC
 116
 117 000044                                   CALL    $SAVAL                    ;SAVE ALL REGISTERS
 118 000050                                   CALL    FNDPDV                    ;FIND ECL/XPT DATA STRUCTURES
 119 000054  005067  177724                   CLR     ACTCIR                    ;INIT CIRCUIT ENABLED FLAG
 120 000060  017746  000000G                  MOV     @SLTNM,-(SP)              ;SET NUMBER OF SYSTEM LINES
 121 000064  017702  000000G                  MOV     @LLCTA,R2                 ;POINT TO FIRST REVERSE MAPPING
 122 000070  017704  000000G                  MOV     @SLTMA,R4                 ;POINT TO SLT INDEX TABLE
 123 000074  012405                   ENAC:   MOV     (R4)+,R5                  ;GET POINTER TO SYSTEM LINE TABL
 124 000076  005067  177676                   CLR     STAADD                    ;ASSUME NOT MULTIPOINT LINE
 125 000102  032765  040000   000000          BIT     #LF.RDY,L.FLG(R5)         ;IS EVERYTHING READY (DDM,DLC,ETC.)?
 126 000110  001002                           BNE     1$                        ;IF YES - ENABLE IT
 127 000112                                   CALLR   60$                       ;CONTINUE
 128
 129 000116  116500  000003           1$:     MOVB    L.DLC(R5),R0              ;GET PDV INDEX OF DDM
 130 000122  067700  000000G                  ADD     @PDVTA,R0                 ;POINT TO CORRECT PDV
 131 000126  011000                           MOV     (R0),R0                   ;GET PDV ADDRESS
 132 000130  032765  100000   000000          BIT     #LF.ACT,L.FLG(R5)         ;IS LINE ALREADY ACTIVE?
 133 000136  001412                           BEQ     5$                        ;IF NO - BRANCH
 134
 135 000140  032765  000400   000000          BIT     #LF.BRO,L.FLG(R5)         ;BROADCAST LINE ?
 136 000146  001404                           BEQ     4$                        ;IF NO - BRANCH
 137 000150  132765  000200   000022          BITB    #SF.ACT,L.MPF(R5)         ;XPT CHANNEL ACTIVE ?
 138 000156  001402                           BEQ     5$                        ;IF NO - BRANCH
 139
 140 000160                           4$:     CALLR   60$                       ;CONTINUE
 141
 142 000164                           5$:
 143                                          .IF DF   R$$MPL
 144
 145                                          BIT     #ZF.PSE,Z.FLG(R0)         ;DLM CIRCUIT ?
 146                                          BNE     20$                       ;IF YES - BRANCH
 147
 148                                          MOV     L.KRBA(R5),R3             ;GET POINTER TO KRB
 149                                          BIT     #KS.OFL,K.STS(R3)         ;IS THE CONTROLLER OFF-LINE?
 150                                          BNE     4$                        ;YES ... CAN'T ENABLE IT
 151                                          TSTB    @NCPU                     ;ARE WE RUNNING ON A MULTI-PROCESSOR?
 152                                          BEQ     20$                       ;NO ... WE CAN ENABLE THE LINE
 153                                          BIT     K.URM(R3),@URMST          ;IS THE REQUIRED UNIBUS ACTIVE?
 154                                          BEQ     4$                        ;NO ... WE CAN'T ENABLE THE LINE
```

```
                                                                   D 12
NTIMFE     CREATED BY  MACRO  ON 24-JUL-85 AT 15:09      PAGE 5

MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME          REFERENCES

ADJDF$              #5-65     5-79
BIAS                #5-66     8-347
CALL                 6-117    6-118     6-193     6-198     6-237     6-241     7-282     7-288     7-289     8-329
                     8-364
CALLR               #5-67     6-127     6-140     6-249     7-297
CCBDF$              #5-65     5-71
CTRDF$              #5-66     5-72
DHBDF$              #5-67     5-69
DIR$                #6-113    6-113    #6-114     6-114    #7-303     7-303
ECDDB$              #5-67     5-70
EMSG$               #5-67     6-193     6-237
MAP                 #5-66     8-344
MOV$                #6-113    6-113     6-113     6-113     6-113    #6-114     6-114
MRKT$S              #5-67     6-113
PDVDF$              #5-65     5-77
PHBDF$              #5-66     5-76
PLBDF$              #5-66     5-78
RESMAP              #5-66     8-371     8-374
RESRG               #5-65     7-307
RETC                #5-65     7-299     8-333
RETURN               6-253    7-300     7-308     8-375
RVP$                #6-113    6-113
SAVMAP              #5-66     8-342     8-359
SAVRG               #5-65     7-269
SLTDF$              #5-65     5-74
SOB                  8-353
SVCDF$              #5-66     5-75
SWSTK$               7-288    8-329
WSIG$S              #5-65     7-303
WTSE$S              #5-67     6-114
XPDDB$              #5-65     5-73
```

```
                                                                   E 12
**FILE**ID**NTIMFL
```

```
                                                                        D 13
NTIMFL      CREATED BY  MACRO  ON 29-JUN-85 AT 03:27      PAGE 3

SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE               REFERENCES

PR3      = 000140            #5-66
PR4      = 000200            #5-66
PR5      = 000240            #5-66
PR6      = 000300            #5-66
PR7      = 000340            #5-66
PS       = 177776            #5-66
R$$MPL   = ******             5-66         5-66         5-66
R$$11D   = ******             5-64         5-66
R$$11M   = 000000             5-64
R$$11S   = ******             5-64
SF.ACT   = 000200            #5-65
SF.ENA   = 000100            #5-65
SF.LPB   = 000004            #5-65
SF.MFL   = 000040            #5-65         9-192        9-203
SF.PAC   = 000020            #5-65
SF.REA   = 000010            #5-65
SF.SER   = 000001            #5-65
SF.SVC   = 000002            #5-65
SF.UNL   = 000040            #5-65
SLTMA    = ******    GX       8-132
SLTNM    = ******    GX       8-131
SWR      = 177570            #5-66
S$$BAS   = ******             6-80         6-80         6-81         6-81
S.COST     000001            #5-65
S.FLG      000000            #5-65         9-192       *9-203
S.LEN      000004            #5-65         9-205
S.NMST     000002            #5-65
S.OWNR     000003            #5-65
TPS      = 177564            #5-66
UBMPR    = 170200            #5-66
UISAR0   = 177640            #5-66
UISAR1   = 177642            #5-66
X$$MCB   = ******             5-64         5-64
ZF.COU   = 001000            #5-64
ZF.DDM   = 000001            #5-64
ZF.DIA   = 004000            #5-64
ZF.DLC   = 000002            #5-64
ZF.DVP   = 100000            #5-64
ZF.INI   = 040000            #5-64
ZF.KMX   = 000020            #5-64
ZF.LLC   = 000004            #5-64
ZF.LMC   = 000100            #5-64
ZF.MAN   = 020000            #5-64
ZF.MFL   = 000010            #5-64         7-115
ZF.MTM   = 000400            #5-64
ZF.MUX   = 000040            #5-64         8-161
ZF.PSE   = 002000            #5-64         8-139
ZF.SLI   = 010000            #5-64
ZF.TIM   = 000200            #5-64
ZF.X3P   = 000000            #5-64
ZS.ASN   = 100000            #5-64
ZS.BSY   = 140000            #5-64
```

```
                                                                        E 13
NTIMFL      CREATED BY  MACRO  ON 29-JUN-85 AT 03:27      PAGE 4

SYMBOL CROSS REFERENCE                                    CREF    04.00
```

```
                                              .SBTTL  MAIN CONTROL

                                      ;+
                                      ; NTIGRQ - GET FUNCTION REQUEST AND DISPATCH
                                      ;
                                      ; THIS ROUTINE IS CURRENTLY CODED SUCH THAT FUNCTION REQUESTS TO
                                      ; NTINIT VIA "SEND DATA" MUST BE SYNCHRONOUS WITH NTINIT
                                      ; COMPLETIONS.  THAT IS, ANY REQUEST TO NTINIT IS ILLEGAL UNTIL
                                      ; THE PREVIOUS REQUEST HAS BEEN COMPLETED.
                                      ;
                                      ; INPUTS:
                                      ;       NONE
                                      ;
                                      ; OUTPUTS:
                                      ;       WAIT FOR A COMPLETED RCV DATA AND DISPATCH THE PROPER FUNCTION
                                      ;         ACCORDING TO "$NIFNC" (SET BY THE RCV DATA ROUTINE)
                                      ;       THE SEND DATA TO THE REQUESTING TASK IS ONLY DONE AFTER
                                      ;         THE FUNCTION IS COMPLETED.  THE RETURNED FUNCTION IS EITHER
                                      ;         POSITIVE (SUCCESS), ZERO (NO-OP), OR NEGATIVE (ERROR).
                                      ;
                                      ;-

000060                                NTIGRQ:
                                              .IF DF  R$$11S
                                              BIT     #FE.CEX,@FMASK          ;IS THE COMM EXEC LOADED ?
                                              BEQ     20$                     ; .. NO
                                              .ENDC

000060  126727  000204' 000001                CMPB    $NIFNC,#LS.CEX          ;IS IT A "SET SYS"?
000066  001424                                BEQ     30$                     ;BR IF YES

                                              .IF NDF R$$11S
000070  032777  000000G 000610'               BIT     #FE.CEX,@FMASK          ;IS THE COMM EXEC LOADED ?
000076  001414                                BEQ     20$                     ; .. NO
000100  126727  000204' 000011                CMPB    $NIFNC,#LS.ON           ;IS IT "SET EXECUTOR ON"?
000106  001430                                BEQ     40$                     ;BR IF YES
000110  126727  000204' 000012                CMPB    $NIFNC,#LS.X25          ;IS IT SET MODULE X25-SERVER?
000116  001452                                BEQ     60$                     ;BR IF YES
                                              .ENDC

                                              .IF DF  S$$BAS
                                              CMPB    $NIFNC,#LS.ON           ;IS IT "SET EXECUTOR STATE ON"?
                                              BEQ     40$                     ;IF EQ, YES
                                              .ENDC

000120                                        EMSG$   03                      ;ILLEGAL FUNCTION
000126  000470                                BR      80$
000130                                20$:    EMSG$   04                      ;COMM EXEC NOT LOADED
000136  000464                                BR      80$
                                      ;
                                      ; CALLS TO FUNCTION ROUTINES.  THESE ROUTINES SHOULD RETURN WITH CARRY
                                      ; INDICATING SUCCESS/FAILURE, AND "$NIFNC" SET TO A PROPER RETURN
                                      ; COMPLETION CODE FOR THE FUNCTION REQUESTOR.
                                      ;
000140                                30$:    CALL    $NICEX                  ;FUNCTION - SET SYS
000144  103461                                BCS     80$                     ;BR IF FAILED
000146  032767  010000  000204'               BIT     #LS.CXO,$NIFNC          ; IS IT EXECUTOR ON ALSO ?
000154  001005                                BNE     40$                     ; BR IF YES
```

```
                                         .SBTTL  PUTAST - PUT COMPLETION AST IN CALLER TASK'S AST QUEUE

                                ;+
                                ; PUTAST - PUT A COMPLETION AST IN THE CALLER TASK'S AST QUEUE
                                ;
                                ; INPUTS:
                                ;        $NITRN=CALLING TASK'S NAME
                                ;        $NIFNC=SUCCESS CODE
                                ;        $NIAST=AST ADDRESS IN TASK
                                ;        $NIPRM=PARAMETER TO RETURN TO TASK
                                ;
                                ; OUTPUTS:
                                ;        AN AST IS QUEUED FOR THE CALLER.
                                ;-

        000000                  $$$$ = 0                                ; ASSUME WE WILL ASSEMBLE THIS ROUTINE
                                         .IIF DF R$$11S  $$$$ = 1       ; WE DON'T NEED IT FOR 11S
                                         .IIF DF S$$BAS  $$$$ = 0       ; BUT WE DO NEED IT FOR SERVER BASE
                                                                        ; (EVEN THOUGH IT IS 11S BASED)

                                         .IF EQ  $$$$
001222                          PUTAST:
001222                                   SWSTK$  30$                    ;; ENTER SYSTEM STATE
001226  012703  000174'                  MOV     #NTISEN+S.DATN,R3      ;; POINT TO TASK NAME
001232                                   CALL    @SRSTD                 ;; FIND CALLER'S TCB ADDRESS
001236  103406                           BCS     10$                    ;; IF CS, NOT THERE - TOO BAD
001240  010005                           MOV     R0,R5                  ;; ELSE, COPY TCB ADDRESS
001242  012701  000016                   MOV     #7*2,R1                ;; ELSE, SET AST BLOCK LENGTH
001246                                   CALL    @ALOCB                 ;; ALLOCATE SOME POOL
001252  103006                           BCC     20$                    ;; IF CC, GOT IT - PROCEED

001254                          10$:     RETC    R0                     ;; RETURN CARRY SET
001266  000421                           BR      30$                    ;; ... AND RETURN

001270  010002                  20$:     MOV     R0,R2                  ;; COPY ADDRESS OF BLOCK ALLOCATED
001272  005720                           TST     (R0)+                  ;; BUMP OVER LINK WORD
001274  010120                           MOV     R1,(R0)+               ;; SET LENGTH OF BLOCK (FOR DEALLOCATION)
001276  012720  000024                   MOV     #12*2,(R0)+            ;; SET STACK SPACE TO RESERVE FOR TASK
001302  016720  000120'                  MOV     $NIAST,(R0)+           ;; SET AST VIRTUAL ADDRESS FOR TASK
001306  012720  000002                   MOV     #2,(R0)+               ;; SET PARAMETER COUNT
001312  016720  000122'                  MOV     $NIPRM,(R0)+           ;; SET PARAMETER #1 (USER SUPPLIED)
001316  016720  000204'                  MOV     $NIFNC,(R0)+           ;; SET PARAMETER #2 (COMPLETION STATUS)
001322  010500                           MOV     R5,R0                  ;; POINT TO TASK TCB
001324  010201                           MOV     R2,R1                  ;; POINT TO AST BLOCK
001326                                   CALL    @QASTT                 ;; QUEUE THE AST TO THE CALLING
001332                          30$:     RETURN                         ;; BACK TO USER STATE AND RETURN

                                         .ENDC


        000000'                          .END    NTINIT
```

Table of contents

```
     5-    55     DATA
     6-    80     START CEX TIMERS
     7-   121     T1SIN - INSERT 1 SECOND TIMER FORK BLOCK
     7-   122     T50IN - INSERT 50 MSEC TIMER FORK BLOCK
     8-   156     CLINS - INSERT ENTRY ON CLOCK QUEUE
```

```
   1635
   1636 005422  024242                   80$:     CMP     -(R2),-(R2)         ; RESET R2 TO FIRST WORD OF ENTRY
   1637 005424  010267  000010'                   MOV     R2,UBA              ; SAVE UNLOAD ENTRY ADDRESS WITHIN BLOCK
   1638 005430  017767  000000G 000006'           MOV     @$ADRX,UBIAS        ;  AND ADDRESS
   1639 005436  012702  000012'                   MOV     #BLOCK,R2           ; RESET R2 TO START OF BLOCK
   1640
   1641 005442                           100$:    SWTCHI                      ; POINT BACK AT DEFAULT BUFFER
   1642 005450                                    RETURN
```

```
VUN
SYM
SYM
Z.D
Z.D
Z.F
Z.N
Z.P
$AD
$AD
$BF

$BI
$CC
$CC
$CC
$CE
$CE
$CL
$CP
$CX
$DD
$DE
$DE
$DE
$DE
$DE
$DE
$DL
$ER
$FI
$FL
$FM
$GE
$GE

$HM
$IC
$KR
$LC
$ML
$MX
$NA
$NT

$OE
$PA
$PD
```

```
VUN
SYM
```

```
                                                                   E  2
VUNL       CREATED BY  MACRO  ON 16-SEP-85 AT 07:31      PAGE 5
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE              REFERENCES

Z.DAT   = ******  GX     8-366     *8-385     10-484
Z.DSP   = ******  GX    *8-361     *8-362     14-745     18-1150
Z.FLG   = ******  GX     8-316      8-364    *13-619     14-655     14-757     14-775
Z.NAM   = ******  GX     8-372      8-377     14-754
Z.PCB   = ******  GX     8-351     *8-360     21-1305    21-1480    24-1813
$ADRX   = ******  GX     18-1093    22-1598   22-1632    22-1638
$ADRZ   = ******  GX     22-1597
$BFR    = ******  GX     7-210      7-215     7-219      7-226     *8-318      8-320      8-323      8-325      8-327
                         8-329     *8-331     8-335      8-337      8-341      8-343      10-443     10-444     10-450
                         10-455     10-458    10-464     10-468     10-477     10-481     10-486     10-487     10-500
                         11-525     11-528    11-532     14-726     15-833     15-845     15-847     15-849     17-986
                         17-992     17-1029   17-1034    17-1047    18-1165    20-1227    21-1265   *21-1267   *21-1270
                         21-1324   *21-1393  *21-1465    21-1467    21-1471    21-1473    21-1475    21-1477    21-1480
                        *21-1482   *21-1483   21-1498    21-1510    21-1518    21-1519    21-1527    21-1539    21-1548
                         21-1568    21-1569   22-1641    23-1749    23-1752   *23-1755    23-1778    24-1818    25-1850
$BIAS   = ******  GX    *14-745    *24-1818
$CCBLH  = ******  GX     21-1321   *21-1324
$CCBSZ  = ******  GX     21-1313
$CCEX     003666 RG     #21-1253
$CEACC  = ******  GX     7-207      21-1494   21-1515    21-1545    21-1565
$CETDF  = ******  GX     7-214      20-1217
$CLIN     001610 RG     #14-648
$CPRO     001462 RG     #13-614
$CXOPT  = ******  GX    *21-1357
$DDMPV  = ******  GX     12-587
$DEA16  = ******  GX     14-808     21-1281   21-1332    21-1361    21-1390    21-1431    24-1826
$DEA18  = ******  GX     10-452     10-465    10-478     10-496     10-502     11-533     23-1711    23-1733
$DECPT  = ******  GX     10-485     21-1342   21-1387    21-1509    21-1538
$DEPOL  = ******  GX     7-234      21-1499   21-1528    21-1552    21-1571
$DEUMR  = ******  GX     21-1429
$DEVHD  = ******  GX     21-1465
$DLCPV  = ******  GX     12-573     12-582
$EROUT  = ******  GX     13-626     17-1065   20-1234
$FILHD  = ******  GX     21-1562
$FLAGS  = ******  GX     8-264      12-571    12-580     12-585
$FMASK  = ******  GX     21-1256   *21-1386
$GETAD  = ******  GX     7-208      18-1117   21-1496    21-1517    21-1547    21-1567    22-1601    23-1748
$GETRV  = ******  GX     8-322      8-334     10-442     10-453     10-466     10-479     10-485     10-497     11-524
                         14-700     15-827    15-836     17-962     17-1007    21-1264    21-1323    21-1469    21-1509
                         21-1538    24-1814   25-1849    25-1856
$HMBUF  = ******  GX     21-1508    21-1512   21-1537    21-1541
$ICBLN  = ******  GX     15-825     18-1090   18-1161
$KRBLN  = ******  GX     14-700     14-802    17-962
$LGSTT  = ******  GX    *21-1575
$MUL    = ******  GX     10-472     10-489
$MXVCT  = ******  GX     17-994     17-1003   17-1007    17-1050    17-1068
$NAME   = ******  GX     8-335      13-615   *14-754
$NTLHB  = ******  GX     7-203      7-224     20-1205    21-1359    21-1360    21-1363    21-1364    21-1366    21-1371
                         21-1379    22-1598   23-1734   *23-1736    23-1741    23-1746
$OBJHD  = ******  GX     21-1491
$PARHD  = ******  GX     8-318
$PDVA   = ******  GX     8-263      13-617   *21-1307
```

```
                                                                   F  2
VUNL       CREATED BY  MACRO  ON 16-SEP-85 AT 07:31      PAGE 6
SYMBOL CROSS REFERENCE                                   CREF   04.00
```

```
                                               .SBTTL  DEACUG - Deallocate CUG blocks                      ;[TD001]
                                       ;+                                                                  ;[TD001]
                                       ; *** DEACUG - Deallocate CUG blocks                                ;[TD001]
                                       ;                                                                   ;[TD001]
                                       ; INPUTS                                                            ;[TD001]
                                       ;       $PFLAG - PSI characteristics word                           ;[TD001]
                                       ;                                                                   ;[TD001]
                                       ; OUTPUTS                                                           ;[TD001]
                                       ;       CUG blocks deallocated                                      ;[TD001]
                                       ;                                                                   ;[TD001]
                                       ;       All registers preserved                                     ;[TD001]
                                       ;-                                                                  ;[TD001]
                                                                                                           ;[TD001]
000374                                 DEACUG: CALL    $SAVAL          ; Save all registers                ;[TD001]
000400  016703  000000G                        MOV     $PSIPT,R3       ; Get address of listhead           ;[TD001]
000404  062703  000010                         ADD     #H$CUG,R3                                           ;[TD001]
000410  032767  000000G 000000G                BIT     #PF.XDF,$PFLAG  ; Check for extended data format    ;[TD001]
000416  001005                                 BNE     10$             ; Branch if extended                ;[TD001]
                                       ;                                                                   ;[TD001]
                                       ; Deallocate old-format CUG blocks                                  ;[TD001]
                                       ;                                                                   ;[TD001]
000420  012701  000016                         MOV     #G$LEN,R1       ; Length of old-format block        ;[TD001]
000424                                         CALL    DEALST          ; Deallocate                        ;[TD001]
000430  000447                                 BR      60$                                                 ;[TD001]
                                       ;                                                                   ;[TD001]
                                       ; Deallocate extended CUG blocks                                    ;[TD001]
                                       ;                                                                   ;[TD001]
000432                                 10$:    SAVMAP                  ; Save current mapping              ;[TD001]
000436                                         GETRV   R3,#2           ; Read in listhead                  ;[TD001]
000454  016746  000000G                        MOV     $BFR,-(SP)      ; Save address                      ;[TD001]
000460  005067  000000G                        CLR     $BFR            ; Zero buffer and rewrite listhead  ;[TD001]
000464                                         PUTRC                                                       ;[TD001]
                                                                                                           ;[TD001]
000470  012600                         20$:    MOV     (SP)+,R0        ; Restore address of block          ;[TD001]
000472  001424                                 BEQ     50$             ; Branch if end of chain            ;[TD001]
000474                                         CEACC$  R0,R2           ; Map to block                      ;[TD001]
000504                                         GETAD   R2,#G$LEN       ; Read fixed part of block          ;[TD001]
000522  016746  000000G                        MOV     $BFR,-(SP)      ; Save link address                 ;[TD001]
000526  116701  000015G                        MOVB    $BFR+G$NML,R1   ; Get length of variable part       ;[TD001]
000532  062701  000016                         ADD     #G$LEN,R1       ; Add length of fixed part          ;[TD001]
000536                                         CALL    $DEFPOL         ; Deallocate block                  ;[TD001]
000542  000752                                 BR      20$                                                 ;[TD001]
                                                                                                           ;[TD001]
000544                                 50$:    RESMAP                  ; Restore mapping                   ;[TD001]
000550                                 60$:    RETURN                                                      ;[TD001]
```

```
                                                                              E  4
VUNPSI     CREATED BY  MACRO  ON 17-JUL-85 AT 12:15       PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

R$R2     = 000006           #6-93
R$R3     = 000010           #6-94
R$R4     = 000012           #6-95      13-414
R$R5     = 000014           #6-96
R$$EIS   = ******            12-356     15-475
R$$11D   = ******            12-356     15-475
SYSFDB   = ******   GX      *9-185     *9-193     *10-231    *10-239    *11-263    *11-276    *11-288    *12-327    *12-330
                            *12-331    *12-335    *12-347    *12-365    *13-385    *13-388    *13-395    *14-436    *15-464
                            *15-469    *16-507    *16-514    *17-539    *17-547
S$LEN      000032            8-125
T$CBUF     000004            15-478
T$CLEN     000020            15-469     15-472
T$CSIZ     000010            15-474
XLDTE      000772 R          8-137     #12-323
XPORT      001260 R          8-136     #13-383
XPSV       001720 R          8-140     #16-501
XTRA       001554 R          8-139     #15-459
XX29       001476 R          8-138     #14-434
X$GLEN     000106            13-405
X$LEN      000106            13-402
$BFR     = ******   GX       9-186     *9-187      9-194      9-195      10-232    *10-233     10-240     10-241     11-264
                            *11-265     11-277     11-279     11-282     11-289     11-294     11-297     12-331     12-336
                             12-345     12-348    *12-348     12-353     12-355    *12-364     13-386     13-389     13-396
                             14-437    *14-439     15-465    *15-467     15-474     15-478     16-508     16-515     17-540
                            *17-541     17-548
$CEACC   = ******   GX       9-192      10-238     11-270     12-334     13-392     17-546
$DBG$    = ******            8-116
$DEA16   = ******   GX       8-153      12-350     14-442     15-487
$DEPOL   = ******   GX       9-197      10-243     11-302     12-361     13-408     13-416     16-511     16-518     17-551
$GETAD   = ******   GX       9-193      10-239     11-276     11-288     12-335     13-395     17-547
$GETRV   = ******   GX       9-185      10-231     11-263     12-330     12-347     13-385     13-388     14-436     15-464
                             15-469     16-507     16-514     17-539
$PFLAG   = ******   GX       8-131      8-146      8-148     *8-154      9-173      10-219     11-271     12-340     13-403
                             16-501
$PSIPT   = ******   GX       8-117      8-141     *8-155      9-171      10-217     12-328     13-383     14-434     15-462
                             16-505     16-512
$PUTRC   = ******   GX       9-188      10-234     11-266     12-365     14-440     15-468     17-542
$RADDR   = ******   GX      *9-185     *9-193     *10-231    *10-239    *11-263    *11-276    *11-288    *12-330    *12-335
                            *12-347    *12-365    *13-385    *13-388    *13-395    *14-436    *15-464    *15-469    *16-507
                            *16-514    *17-539    *17-547
$RBLCK   = ******   GX       9-184     *9-200      10-230    *10-246     11-262    *11-305     12-325    *12-366     13-406
                            *13-410     17-538    *17-555
$SAVAL   = ******   GX       9-170      10-216     11-261     12-324     13-391     15-460     16-504     17-537
$XPSI      000000 RG        #8-116
```

```
                                                                              F  4
VUNPSI     CREATED BY  MACRO  ON 17-JUL-85 AT 12:15       PAGE 3
MACRO CROSS REFERENCE                                     CREF   04.00
```

```
                                                                     E  5
VVCT        CREATED BY  MACRO  ON 29-JUN-85 AT 03:12      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

$TMRDY     000434 RG         7-254       #8-268
$TMVCT     000000 RG        #6-86
$VECT1   = ******   GX       6-99
$$VECT   = ******   GX       6-120
```

```
                                                                     F  5
VVCT        CREATED BY  MACRO  ON 29-JUN-85 AT 03:12      PAGE 3
MACRO CROSS REFERENCE                                     CREF   04.00
```

```
                                        ;+
                                        ; $XLBR - RELEASE LIBRARY REFERENCE
                                        ;
                                        ; INPUTS:
                                        ;       R0=LIBRARY DESCRIPTOR BIAS
                                        ;       R2=LIBRARY DESCRIPTOR ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       R0,R1,R2=DESTROYED
                                        ;-
000000                                  $XLBR::
000000  005046                                  CLR     -(SP)           ; NO LIBRARY TO DE-ALLOCATE
000002  005046                                  CLR     -(SP)           ; NO DE-ALLOCATION SIZE
000004                                          CALL    $GETLB          ; READ IN LIBRARY BLOCK
000010  060201                                  ADD     R2,R1           ; POINT AT OUR DESCRIPTOR
000012  005767  000000G                         TST     .MGMGE          ; MAPPED TARGET SYSTEM?
000016  100403                                  BMI     10$             ; IF MI, NO
000020  105361  000000G                         DECB    LB.USE(R1)      ; DECREMENT USE COUNT
000024  000402                                  BR      20$
000026  005361  000000G                 10$:    DEC     LB.USE(R1)      ; DECREMENT USE COUNT
000032  001046                          20$:    BNE     50$             ; IF NE, STILL SOME ACTIVE REFERENCES
000034  005011                                  CLR     (R1)            ; ZERO THE FIRST WORD OF DESCRIPTOR
000036  016116  000000G                         MOV     LB.ADR(R1),(SP) ; SAVE LIBRARY BIAS AND SIZE FOR LATER
000042  016166  000000G 000002                  MOV     LB.SIZ(R1),2(SP) ;
000050  160201                                  SUB     R2,R1           ; POINT AT START OF 32. WORD BLOCK
000052  026700  000000G                         CMP     $NTLPT,R0       ; WAS DESCRIPTOR WITHIN NETLDR DATA BLOCK?
000056  001434                                  BEQ     50$             ; IF EQ, YES
000060  005361  000002                          DEC     2(R1)           ; ADJUST DESCRIPTOR COUNT FOR THIS BLOCK
000064  001031                                  BNE     50$             ; IF NE, BLOCK IS STILL ACTIVE
000066  011146                                  MOV     (R1),-(SP)      ; SAVE POINTER TO NEXT BLOCK
000070  010046                                  MOV     R0,-(SP)        ; SAVE BIAS OF THIS BLOCK
000072  012701  000001                          MOV     #1,R1           ; ASSUME MAPPED TARGET
000076  005767  000000G                         TST     .MGMGE          ; CORRECT ASSUMPTION?
000102  100002                                  BPL     25$             ; IF PL, YES
000104  012701  000100                          MOV     #100,R1         ; UNMAPPED ALLOCATION SIZE
000110                                  25$:    CALL    $DEA18          ; DE-ALLOCATE LIBRARY BLOCK
000114  012602                                  MOV     (SP)+,R2        ; RESTORE OUR BIAS
000116  016700  000000C                         MOV     $NTLHB+.CXLBR,R0 ; GET FIRST LIBRARY BLOCK BIAS
000122  020002                                  CMP     R0,R2           ; IS OURS THE FIRST?
000124  001004                                  BNE     40$             ; IF NE, NO
000126  012667  000000C                         MOV     (SP)+,$NTLHB+.CXLBR ; SET NEW FIRST LIBRARY BLOCK BIAS
000132  000410                                  BR      55$
000134  011100                          30$:    MOV     (R1),R0         ; GET NEXT LIBRARY BLOCK BIAS
000136                                  40$:    CALL    $GETLB          ; READ IN LIBRARY BLOCK
000142  021102                                  CMP     (R1),R2         ; DOES THIS BLOCK POINT AT OURS?
000144  001373                                  BNE     30$             ; IF NE, NO
000146  012611                                  MOV     (SP)+,(R1)      ; RESET POINTER TO OUR VALUE
000150                                  50$:    CALL    $PUTLB          ; RE-WRITE LIBRARY BLOCK
000154  012600                          55$:    MOV     (SP)+,R0        ; GET LIBRARY BIAS
000156  012601                                  MOV     (SP)+,R1        ; AND SIZE
000160  001402                                  BEQ     60$             ; IF ZERO, ALL DONE

                                                .IF DF  R$$MPL                                                  ;[MP01]
                                                CALL    DELIBV          ; DEALLOCATE LIBRARY VECTOR BLOCK       ;[MP01]
                                                .ENDC                                                           ;[MP01]

000162                                          CALL    $DEA18          ; DE-ALLOCATE LIBRARY
```

E 7

```
NNN         NNN     TTTTTTTTTTTTTTT     IIIIIIIII        111         111        MMM         MMM
NNN         NNN     TTTTTTTTTTTTTTT     IIIIIIIII        111         111        MMM         MMM
NNN         NNN     TTTTTTTTTTTTTTT     IIIIIIIII        111         111        MMM         MMM
NNN         NNN           TTT              III        111111      111111        MMMMMM   MMMMMM
NNN         NNN           TTT              III        111111      111111        MMMMMM   MMMMMM
NNN         NNN           TTT              III        111111      111111        MMMMMM   MMMMMM
NNNNNN      NNN           TTT              III           111         111        MMM   MMM   MMM
NNNNNN      NNN           TTT              III           111         111        MMM   MMM   MMM
NNNNNN      NNN           TTT              III           111         111        MMM   MMM   MMM
NNN   NNN   NNN           TTT              III           111         111        MMM         MMM
NNN   NNN   NNN           TTT              III           111         111        MMM         MMM
NNN   NNN   NNN           TTT              III           111         111        MMM         MMM
NNN      NNNNNN           TTT              III           111         111        MMM         MMM
NNN      NNNNNN           TTT              III           111         111        MMM         MMM
NNN      NNNNNN           TTT              III           111         111        MMM         MMM
NNN         NNN           TTT              III           111         111        MMM         MMM
NNN         NNN           TTT              III           111         111        MMM         MMM
NNN         NNN           TTT              III           111         111        MMM         MMM
NNN         NNN           TTT           IIIIIIIII     111111111   111111111     MMM         MMM
NNN         NNN           TTT           IIIIIIIII     111111111   111111111     MMM         MMM
NNN         NNN           TTT           IIIIIIIII     111111111   111111111     MMM         MMM
```

F 7

```
**FILE**ID**NTICEX
```

```
                                                                  E  8
NTICEX     CREATED BY  MACRO  ON 29-JUN-85 AT 03:25      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE              REFERENCES

L.DLM      000006           #5-73
L.DLS      000010           #5-73
L.FLG      000000           #5-73
L.KRBA     000016           #5-73
L.LEN    = 000022           #5-73
L.MPF      000022           #5-73
L.NMST     000020           #5-73
L.NSTA     000014           #5-73
L.OWNR     000021           #5-73
L.UNT      000013           #5-73
MPAR     = 172100           #5-74
MPCSR    = 177746           #5-74
M$$EXT   = 000000   G       #5-78          7-230       7-237
M$$MGE   = 000000            5-74
NTILIN     000110 RG         6-138        #7-188
NTLPT    = ******   GX       7-228
PDVID    = ******   GX       7-204
PDVNM    = ******   GX       7-191
PDVTA    = ******   GX       7-190
PIRQ     = 177772           #5-74
PMODE    = 030000           #5-74
PR0      = 000000           #5-74
PR1      = 000040           #5-74
PR2      = 000100           #5-74
PR3      = 000140           #5-74
PR4      = 000200           #5-74
PR5      = 000240           #5-74
PR6      = 000300           #5-74
PR7      = 000340           #5-74
PS       = 177776           #5-74
PUMR     = ******   GX       7-260
R$$MPL   = ******            5-67          5-74        5-74        5-74        5-96        7-222       7-242       7-254       7-273
                             8-300
R$$11D   = ******            5-74          5-75
R$$11M   = 000000            5-75
R$$11S   = ******            5-75          5-88        6-135       7-216
SF.ACT   = 000200           #5-73
SF.ENA   = 000100           #5-73
SF.LPB   = 000004           #5-73
SF.MFL   = 000040           #5-73
SF.PAC   = 000020           #5-73
SF.REA   = 000010           #5-73
SF.SER   = 000001           #5-73
SF.SVC   = 000002           #5-73
SF.UNL   = 000040           #5-73
SWR      = 177570           #5-74
S$$BAS   = ******            5-90          5-90
S.COST     000001           #5-73
S.FLG      000000           #5-73
S.LEN      000004           #5-73
S.NMST     000002           #5-73
S.OWNR     000003           #5-73
```

```
                                                                  F  8
NTICEX     CREATED BY  MACRO  ON 29-JUN-85 AT 03:25      PAGE 3
```

E 9

```
NTICSR     CREATED BY  MACRO  ON 29-JUN-85 AT 03:25      PAGE 4

MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME       REFERENCES

CALL              6-123       6-179       6-189
DIR$             #5-59        6-149       6-151
EMSG$            #5-59        6-189
NHWDF$           #5-59        5-61
NTLER$           #5-59        5-88
OFF$             #5-74       #5-75
PDVDF$           #5-59        5-63
RESRG            #5-59        6-190       6-198
RETURN            6-143       6-199
SAVRG            #5-59        6-119       6-161
SLTDF$           #5-59        5-62
SVTK$            #5-59        5-74        5-75
SWSTK$            6-123
```

F 9

```
**FILE**ID**NTILON
```

```
Symbol table

H.EFLM  000004              NF$SHU= 000004              N$VER   000066              PT$DRT= 000100              R.UNRM= 000022
H.EFSV  000040              NF$TIM= 000200              N$XLEN  000124              PT$END= 000004              R.UNRU= 000024
H.FCS   000050              NF.ACC= 000001              N$$ACC= 000001              PT$LV1= 000002              R.UNSM= 000016
H.FORT  000052              NF.BCP= 000040              N$$BUF= 000001              PT$LV2= 000001              R.UNSU= 000020
H.FPSA  000042              NF.END= 000200              N$$LDV= 000001              PT$PH3= 000010              R.UNTN= 000002
H.FPSL= 000062              NF.EVT= 000002              N$$MCP= 000001              PT$XAR= 000020              SRSTD = ****** GX
H.FPVA  000034              NF.LLI= 000004              N$$MLL= 000001              PX$BLK= 000040              SWR   = 177570
H.GARD  000072              NF.LV2= 000100              N$$MOV= 000010              PX$DLM= 000200              S$$WRG= 000000
H.HDLN  000002              NF.SLI= 000400              N$$NCT= 000001              PX$SVC= 000100              S$$YSZ= 007600
H.IPC   000016              NF.SMC= 000020              N$$PEM= 000001              P$CHN   000004              S1.DEC= 001000
H.IPS   000014              NF.SWP= 000010              PDVID = ****** GX           P$CNT   000005              S1.DPR= 000400
H.ISP   000020              N$ACQ   000000              PDVTA = ****** GX           P$CTR   000034              S1.DSI= 004000
H.LUN   000076              N$ACTL  000032              PF$BLK= 020000              P$DRCT  000015              S1.ESC= 000002
H.NLUN  000074              N$ADJ1  000072              PF$CLC= 010000              P$DRTR  000036              S1.IBF= 002000
H.NML   000061              N$ADJ2  000074              PF$DLM= 100000              P$FLG   000012              S1.IBY= 000200
H.ODVA  000022              N$CACH  000062              PF$EIP= 000002              P$FWD   000030              S1.OBY= 000100
H.ODVL  000024              N$CIR   000034              PF$ENB= 000001              P$ICCB  000046              S1.PTH= 000010
H.OVLY  000054              N$CRC   000120              PF$FAI= 004000              P$IPL   000014              S1.RES= 010000
H.PFVA  000032              N$DLA   000020              PF$FM1= 000100              P$LCD   000002              S1.RNE= 000020
H.RCVA  000036              N$DLY   000014              PF$FM2= 000200              P$LEN   000052              S1.RNF= 020000
H.RRVA  000062              N$ELEN  000054              PF$OFF= 000000              P$LST   000000              S1.RSP= 000004
H.SPRI  000060              N$ENC   000042              PF$RM1= 000020              P$NRNI  000040              S1.RST= 000001
H.TKVA  000026              N$ERRC  000022              PF$RM2= 000040              P$OCCB  000050              S1.TNE= 040000
H.TKVL  000030              N$FLG   000005              PF$RVR= 000010              P$PFQ   000006              S1.TSY= 000040
H.VEXT  000056              N$FNC   000006              PF$STA= 000004              P$PKSZ  000044              S1.USI= 100000
H.WND   000044              N$GENQ  000052              PF$SVC= 040000              P$RMX1  000016              S2.BEL= 020000
H.X25   000064              N$GTM   000015              PF$UP = 000004              P$RMX2  000020              S2.BRQ= 000020
I$$RAR= 000000              N$HC1   000052              PIRQ  = 177772              P$RPRI  000042              S2.CR = 002000
I$$RDN= 000000              N$HC2   000056              PLDADR  000000R         002 P$RTIM  000003              S2.CTO= 040000
KISAR0= 172340              N$HIGH  000033              PLDLEN  000002R         002 P$STA1  000022              S2.CTS= 100000
KISAR6= 172354              N$LLT   000026              PMODE = 030000              P$STA2  000023              S2.ELF= 001000
K$$CNT= 177546              N$LLTM  000024              PR$BED= 000200              P$TIM   000010              S2.FLF= 000400
K$$CSR= 177546              N$LVC   000036              PR$BEU= 000100              P$TSCT  000026              S2.IRQ= 000200
K$$LDC= 000000              N$LV1   000002              PR$BRD= 000040              P$TSIZ  000024              S2.OBF= 004000
K$$TPS= 000074              N$LV2   000010              PR$BRU= 000020              P$TYP   000001              S2.ORQ= 000100
LD$LP = 000000              N$MBXQ  000050              PR$DWN= 000002              P$$P45= 000000              S2.PCU= 010000
LDNACP  000014R         002 N$MHC1  000036              PR$LCC= 000010              P$$WRD= 000000              S2.RCU= 000001
LVL1  = 002000              N$MHC2  000044              PR$MOP= 000004              Q$$OPT= 000010              S2.SRQ= 000040
L$$ASG= 000000              N$PLD   000016              PR$UP = 000001              RCP1NM  000004R         002 S2.WAL= 000010
L$$DRV= 000000              N$PLLT  000030              PR0   = 000000              RCP2NM  000010R         002 S2.WRA= 000006
L$$P11= 000001              N$PRI   000076              PR1   = 000040              RF.CTL= 000003              S2.WRB= 000002
L$$11R= 000000              N$ROA1  000022              PR2   = 000100              RF.LD1= 000040              S3.ACR= 000001
MPAR  = 172100              N$ROA2  000030              PR3   = 000140              RF.LD2= 000100              S3.CTC= 000004
MPCSR = 177746              N$RTMX  000014              PR4   = 000200              RF.TIM= 177400              S3.FDX= 000200
M$$CRB= 000124              N$RTM1  000014              PR5   = 000240              RF.TMO= 000400              S3.ICE= 001000
M$$CRX= 000000              N$RTM2  000015              PR6   = 000300              RF.WFC= 000200              S3.MHE= 000400
M$$FCS= 000000              N$RT1   000000              PR7   = 000340              RF.WTD= 000020              S3.NEC= 000020
M$$MGE= 000000              N$RT2   000006              PS    = 177776              RF.WTM= 000030              S3.PPT= 020000
M$$NET= 000000              N$SLA   000016              PS$CHR= 000016              RF.WTS= 000010              S3.PTH= 004000
M$$OVR= 000000              N$SNOD  000012              PS$FAI= 000014              R$$DER= 000000              S3.RAL= 000010
NF$ACC= 100000              N$TCTL  000112              PS$NTI= 000006              R$$K11= 000001              S3.RES= 010000
NF$BLK= 000100              N$TIM   000004              PS$OFF= 000000              R$$SND= 000000              S3.RUB= 040000
NF$DMO= 000010              N$TLC   000100              PS$STR= 000002              R$$11M= 000000              S3.TAB= 000002
NF$EVT= 040000              N$TNC   000114              PS$UP = 000012              R.UNGC= 000015              S3.TME= 002000
NF$MOU= 000040              N$TRC   000106              PS$VER= 000010              R.UNPC= 000014              S3.TSY= 000040
NF$RST= 000002              N$TTCB  000110              PS$WT = 000004              R.UNPN= 000006              S3.8BC= 000100
NF$SCN= 000020              N$VCB   000010              PT$BRO= 000200              R.UNPR= 000012              S4.ABD= 100000
```

```
Symbol table
```

```
                                20$:
                                        .ENDC

000164  116503  000014                  MOVB    L.NSTA(R5),R3           ;GET # OF STATIONS ON THIS LINE
000170  001516                          BEQ     50$                     ;IF EQ, NOT A MULTI-DROP LINE
000172  010246                          MOV     R2,-(SP)                ;SAVE REVERSE MAPPING TABLE POINTER
000174  011202                          MOV     (R2),R2                 ;GET POINTER TO REVERSE MAPPING TABLE/2
000176  006302                          ASL     R2                      ;GET REAL POINTER
000200  005046                          CLR     -(SP)                   ;INITIALISE BIT TO SET IN LINE FLAGS
000202  010546                          MOV     R5,-(SP)                ;SAVE SLT POINTER
000204  010346                          MOV     R3,-(SP)                ;SAVE COUNT OF STATIONS
000206  116567  000012  177567          MOVB    L.CTL(R5),CTLNUM        ;SAVE CONTROLLER NUMBER
000214  105067  177562                  CLRB    STANUM                  ;START WITH STATION NUMBER ZERO
000220  062705  000022                  ADD     #L.MPF,R5               ;POINT TO MULTI-DROP FLAGS TABLE
000224  132715  000100          30$:    BITB    #SF.ENA,(R5)            ;SHOULD THIS STATION BE ENABLED?
000230  001461                          BEQ     40$                     ;NO

000232  011246                          MOV     (R2),-(SP)              ;GET REVERSE MAPPING VALUE
000234  000316                          SWAB    (SP)                    ;GET OWNER LLC INDEX
000236  005001                          CLR     R1                      ; ...
000240  111601                          MOVB    (SP),R1                 ; ...
000242  005726                          TST     (SP)+                   ;CLEAN UP STACK
000244  067701  000000G                 ADD     @PDVTA,R1               ;GET OWNER LLC PDV ADDRESS
000250  011101                          MOV     (R1),R1                 ; ...
000252  022761  114224  000004          CMP     #^RXPT,Z.NAM(R1)        ;OWNER XPT ?
000260  001030                          BNE     36$                     ;IF NO - BRANCH

000262  016701  000000G                 MOV     $DBECL,R1               ;COPY ECL DATA BASE ADDRESS
000266  132761  000040  000005          BITB    #NF$MOU,N$FLG(R1)       ;IS NETWORK ON ?
000274  001001                          BNE     31$                     ;IF YES - BRANCH
000276  000436                          BR      40$                     ;IF NO - DO NOT ENABLE CIRCUIT

000300  017701  000000G         31$:    MOV     @DECPT,R1               ;GET DECNET HOME BLOCK ADDRESS
000304  005761  000060                  TST     D$NN(R1)                ;ROUTING NODE ?
000310  001014                          BNE     36$                     ;IF YES - BRANCH
000312  005767  177466                  TST     ACTCIR                  ;IF NO - ANY OTHER LINES ENABLED ?
000316  001407                          BEQ     35$                     ;IF NO - BRANCH
000320  142765  000100  000000          BICB    #SF.ENA,S.FLG(R5)       ;IF YES - DISABLE LINE
000326                                  EMSG$   12                      ;PRINT MSG
000334  000417                          BR      40$                     ;CONTINUE

000336  005267  177442          35$:    INC     ACTCIR                  ;COUNT ANOTHER XPT CIRCUIT ENABLED
000342  010567  177432          36$:    MOV     R5,STAADD               ;SAVE STATION TABLE ADDRESS
000346                                  CALL    ENALIN                  ;ENABLE THE LINE
000352  103410                          BCS     40$                     ;BR IF LINE NOT OWNED BY SPECIFIED LLC
000354  012766  100000  000004          MOV     #LF.ACT,4(SP)           ;WE WILL MAKE THE LINE ACTIVE
000362  152715  000200                  BISB    #SF.ACT,(R5)            ;MARK THE STATION ACTIVE
000366  112765  000000  000002          MOVB    #LN.ON,S.NMST(R5)       ;MARK THE MANAGEMENT STATE AS 'ON'
000374  022525                  40$:    CMP     (R5)+,(R5)+             ;MOVE ON TO NEXT STATION
000376  062702  000002                  ADD     #2,R2                   ;MOVE ON TO NEXT LINE
000402  105267  177374                  INCB    STANUM                  ;INCREMENT STATION NUMBER
000406  005316                          DEC     (SP)                    ;REDUCE COUNT OF STATIONS TO GO
000410  001305                          BNE     30$                     ;LOOP TILL ALL CHECKED
000412  005726                          TST     (SP)+                   ;REMOVE COUNT FROM STACK
000414  012605                          MOV     (SP)+,R5                ;RECOVER SLT ADDRESS
000416  052665  000000                  BIS     (SP)+,L.FLG(R5)         ;POSSIBLY MARK THE LINE ACTIVE
000422  012602                          MOV     (SP)+,R2                ;RECOVER REVERSE MAPPING TABLE ADDRESS
```

```
**FILE**ID**NTIMFL




NN      NN  TTTTTTTTTT    IIIIII    MM      MM  FFFFFFFFFF  LL
NN      NN  TTTTTTTTTT    IIIIII    MM      MM  FFFFFFFFFF  LL
NN      NN      TT          II      MMMM  MMMM  FF          LL
NN      NN      TT          II      MMMM  MMMM  FF          LL
NNNN    NN      TT          II      MM  MM  MM  FF          LL
NNNN    NN      TT          II      MM  MM  MM  FF          LL
NN  NN  NN      TT          II      MM      MM  FFFFFFFF    LL
NN  NN  NN      TT          II      MM      MM  FFFFFFFF    LL
NN    NNNN      TT          II      MM      MM  FF          LL
NN    NNNN      TT          II      MM      MM  FF          LL
NN      NN      TT          II      MM      MM  FF          LL              ....
NN      NN      TT          II      MM      MM  FF          LL              ....
NN      NN      TT        IIIIII    MM      MM  FF          LLLLLLLLLL      ....
NN      NN      TT        IIIIII    MM      MM  FF          LLLLLLLLLL      ....




LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

```
NTIMFL  MACRO V05.03b Saturday 29-Jun-85 03:27
Table of contents
```

```
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE              REFERENCES

Z.AVL      000014          #5-64
Z.DAT      000016          #5-64
Z.DSP      000000          #5-64      5-64
Z.FLG      000010          #5-64      7-115      8-139      8-151
Z.LEN    = 000016          #5-64
Z.LLN      000006          #5-64
Z.MAP      000020          #5-64
Z.NAM      000004          #5-64      7-121      8-150      9-194
Z.PCB      000012          #5-64
Z.SCH      000007          #5-64
$ERR30     000002 R        #6-80      8-165
$ERR31     000042 R        #6-81      7-127
$NERDD   = ******    GX     6-80
$NER50   = ******    GX     6-81
$NIERA   = ******    GX    *7-126    *8-156     *8-157     *8-158     *8-159     *8-160     *8-163
$NIERR   = ******    GX     6-80      6-81
$NIFLG   = ******    GX     7-118     8-145      9-196
$NIMFL     000000 RG       #7-110
$NINTL   = ******    GX     7-122     8-152      9-201
$NTLFC   = ******    GX    *7-117    *7-120     *7-121      7-124      7-126     *8-144     *8-147     *8-148     *8-149
                           *8-150     8-154      8-156      8-158      8-160     *9-194     *9-195     *9-198     *9-199
                           *9-200
$SAVRG   = ******    GX     9-187
.BASEB   = 140000          #5-66
```

```

                                                .IF NDF R$$11S
000156  032767  000400  000204'                 BIT     #LS.CX5,$NIFNC          ; IS IT SET X25 ON ALSO?
000164  001027                                  BNE     60$                     ; BR IF YES
                                                .ENDC

000166  000454                                  BR      90$                     ; ELSE FINISH UP

000170  105777  000676'                 40$:    TSTB    @LGSTT                  ;ARE ANY LOGGING SINKS ENABLED?
000174  001413                                  BEQ     50$                     ;IF EQ, NO
000176                                          DIR$    #$NIMRK                 ;WAIT TWO SECONDS FOR EVL
000204                                          WTSE$S  #MRKTEF                 ;...
000216                                          DIR$    #$NIEVC                 ;TRY TO START EVENT COLLECTOR
000224                                  50$:    CALL    $NILON                  ;FUNCTION - SET EXECUTOR ON
000230  103427                                  BCS     80$                     ;BR IF ERROR

                                                .IF NDF S$$BAS                  ; DON'T PERMIT UNFIX ON SERVER BASE
                                                  .IF DF R$$11S
                                                BIT     #LS.UNF,$NIFNC          ; IS THE UNFIX SUB-FUNCTION SET ?
                                                BEQ     90$                     ; .. NO
                                                MOV     @TKTCB,R0               ; GET OUR TCB ADDRESS
                                                BIC     #T2.FXD,T.ST2(R0)       ; CLEAR THE FIXED BIT
                                                MOV     #^R...,T.NAM+2(R0)      ; CHANGE INSTALLED NAME TO 'NTI..."
                                                  .IFF  ;DF R$$11S
000232  032767  000400  000204'                 BIT     #LS.CX5,$NIFNC          ; IS IT X25 STATE ON ALSO?
000240  001001                                  BNE     60$                     ; BR IF YES
                                                  .ENDC ;DF R$$11S
                                                .ENDC   ;NDF S$$BAS

000242  000426                                  BR      90$                     ;ELSE EXIT WITH SUCCESS

                                                .IF NDF R$$11S
000244  105777  000676'                 60$:    TSTB    @LGSTT                  ;ARE ANY LOGGING SINKS ENABLED?
000250  001413                                  BEQ     70$                     ;IF EQ, NO
000252                                          DIR$    #$NIMRK                 ;WAIT TWO SECONDS FOR EVL
000260                                          WTSE$S  #MRKTEF                 ;...
000272                                          DIR$    #$NIEVC                 ;TRY TO START EVENT COLLECTOR
000300                                  70$:    CALL    $NIX25                  ;FUNCTION - SET MODULE X25-SERVER
000304  103401                                  BCS     80$                     ;BR IF ERROR
000306  000404                                  BR      90$                     ;FINISH UP

                                                .ENDC
                                        ..
000310  012767  177777  000204'         80$:    MOV     #-1,$NIFNC              ;INDICATE NTINIT FUNCTION FAILED
000316  000402                                  BR      100$                    ;AND FINISH UP
000320  005067  000204'                 90$:    CLR     $NIFNC                  ;INDICATE SUCCESS

000324                                  100$:   .IF NDF R$$11S
000324                                          DSAR$S                          ;DISABLE AST RECOGNITION
000332  005067  000226'                         CLR     $NIFLG                  ;CLR FLAGS AND BUSY INDICATION
000336  005767  000120'                         TST     $NIAST                  ;NOTIFY USERS VIA AST?
000342  001403                                  BEQ     110$                    ;BR IF NO
000344                                          CALL    PUTAST                  ;ELSE NOTIFY USER VIA AST
000350  000403                                  BR      120$                    ;AND CONTINUE

000352                                  110$:   DIR$    #NTISEN                 ;SEND COMPLETION TO THE REQUESTOR
000360                                  120$:   CALL    NTIRDX                  ;GET NEXT FUNCTION OR EXIT
```

```
                                                  E 15
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:02  Page 17-1
Symbol table

ABTIM   000640RG  002 CF.SYN= 000040        C.FLG2  000032        FS.LTM= 001000        LF.X2P= 000000
ALOCB   000620RG  002 CF.TIM= 000400        C.FNC   000010        FS.MNT= 004000        LGSTT   000676RG  002
AS.DEL= 000010        CF.TRN= 000100        C.LIN   000006        FS.MSN= 014000        LLCAL   000706RG  002
AS.EXT= 000004        CLKHD   000634RG  002 C.LNK   000000        FS.REA= 001000        LLCRS   000666RG  002
AS.RED= 000001        CMODE = 140000        C.MOD   000011        FS.REI= 000000        LLCTA   000710RG  002
AS.WRT= 000002        CM.CIR= 000002        C.NSP   000004        FS.REZ= 003000        LN.CLO= 000000
A$$CHK= 000000        CM.FMT= 100000        C.PRO   000042        FS.RLB= 002000        LN.DUM= 000005
A$$CPS= 000000        CM.HRD= 000002        C.RSV   000002        FS.RNG= 011000        LN.LOA= 000004
A$$PRI= 000000        CM.LIN= 000000        C.STA   000007        FS.RST= 000000        LN.LOO= 000003
A$$TRP= 000000        CM.LOO= 000001        C.STS   000012        FS.RTN= 001000        LN.OAU= 000003
A.IOC   000003        CM.XLO= 000004        C.URM   177776        FS.SET= 005000        LN.OFF= 000001
A.LGTH= 000014        CP.DCF= 000040        C.XACP  000004        FS.SFC= 005000        LN.ON = 000000
A.MPCT  000011        CP.HDL= 000007        C.XID   000035        FS.SFR= 006000        LN.OOP= 000004
A.PCB   000012        CP.PS = 177400        C.XLEN  000044        FS.SFS= 004000        LN.OPE= 000001
A.PCBL  000000        CP.PSI= 000200        C.XPLI  000040        FS.SPW= 040000        LN.REF= 000002
A.PRI   000002        CP.XCF= 000100        C.XPT   000034        FS.STM= 000000        LN.SER= 000002
A.STAT  000010        CP.2FR= 000030        C.XSVC  000042        FS.STP= 002000        LN.STA= 000017
A.TCB   000004        CS.ABO= 000100        C.XTC   000037        FS.STR= 001000        LN.SUB= 000360
A.TCBL  000006        CS.BRO= 000002        C.X25   000036        FS.TRM= 003000        LN.TRI= 000006
BUFUMP= 172354        CS.BUF= 000200        DEACB   000622RG  002 FS.WLB= 001000        LO.CON  000006 G
CALLX   000654RG  002 CS.CES= 000002        DECPT   000700RG  002 FS.XKL= 002000        LO.CSR  000006 G
CB.CCB= 000002        CS.CHN= 000010        DEVHD   000642RG  002 FS.XOF= 010000        LO.CTL  000014 G
CB.DDM= 000040        CS.CMP= 000200        D$$BUG= 177514        FS.XON= 007000        LO.LIN  000004 G
CB.DLC= 000020        CS.DCR= 000400        D$$ISK= 000000        FS.ZER= 002000        LO.PAR  000010 G
CB.RDB= 000004        CS.DEF= 000004        D$$L11= 000001        F$$LVL= 000001        LO.PRI  000010 G
CB.SDB= 000010        CS.DEV= 000002        D$$YNC= 000000        G$$TPP= 000000        LO.SCR  000014 G
CB.SLI= 000100        CS.DIS= 000040        D$$YNM= 000000        G$$TSS= 000000        LO.STA  000015 G
CB.XLB= 000001        CS.ENA= 000001        EXEVEC  000606R   002 G$$TTK= 000000        LO.VCT  000012 G
CCBGT   000664RG  002 CS.ENB= 000020        E$$XPR= 000000        G$$WRD= 000000        LR.AST  000026 G
CC.LLC= 000200        CS.ERR= 100000        FC.CCP= 000020        HEADR   000626RG  002 LR.CIR  000002 G
CEACC   000672RG  002 CS.FTL= 001000        FC.CTL= 000006        IE.ACT= ****** GX     LR.CTL  000004 G
CE.ABO= 100362        CS.HCR= 000001        FC.KCP= 000016        IO.WVB= ****** GX     LR.EFN  000004 G
CE.DAO= 100346        CS.HFE= 002000        FC.KIL= 000004        I$$RAR= 000000        LR.LIN  000002 G
CE.DIS= 100366        CS.LST= 040000        FC.MAN= 000024        I$$RDN= 000000        LR.PRM  000030 G
CE.ERR= 100370        CS.MTL= 004000        FC.MLD= 000026        KISAR0= 172340        LR.PRO  000002 G
CE.ILN= 100350        CS.RNG= 000010        FC.PCT= 000030        KISAR6= 172354        LR.STS  000000 G
CE.LTO= 100356        CS.ROV= 000004        FC.PWR= 000022        KSAR6   000612RG  002 LR.TCB  000002 G
CE.MOP= 100372        CS.RSN= 010000        FC.RCE= 000002        K$$CNT= 177546        LR.UNT  000005 G
CE.NTE= 100361        CS.SHU= 000001        FC.RCP= 000014        K$$CSR= 177546        LS.AST= 100000 G
CE.RTE= 100376        CS.SID= 000002        FC.TIM= 000010        K$$LDC= 000000        LS.CEX= 000001 G
CE.SRC= 100364        CS.STR= 000004        FC.XCP= 000012        K$$TPS= 000074        LS.CIR= 000010 G
CE.STP= 100352        CS.SUC= 000001        FC.XME= 000000        LDPWF   000644RG  002 LS.CXO= 010000 G
CE.TME= 100354        CS.TMO= 020000        FE.CEX= ****** GX     LD$LP = 000000        LS.CX5= 000400 G
CE.TMO= 100374        CS.XUR= 000004        FMASK   000610RG  002 LF.ACT= 100000        LS.DLM= 004000 G
CE.UNS= 100344        C$$CKP= 000000        FNDPDV  000450RG      LF.BRO= 000400        LS.ECH= 001000 G
CF.CHN= 000001        C$$ORE= 000400        FS.AST= 000000        LF.BWT= 000007        LS.FDX= 004000 G
CF.DDM= 000002        C$$RSH= 177564        FS.CIB= 002000        LF.ENA= 002000        LS.HDX= 010000 G
CF.DYN= 000004        C.ADD   000034        FS.CRA= 001000        LF.LPB= 001000        LS.LIN= 000003 G
CF.EIS= 000010        C.BID   000003        FS.DIS= 013000        LF.MDC= 000100        LS.LMC= 000007 G
CF.EOM= 000004        C.BUF   000014        FS.DVC= 001000        LF.MFL= 004000        LS.NTI= 004000 G
CF.FRK= 100000        C.BUF1  000014        FS.ENB= 012000        LF.MTP= 000020        LS.ON = 000011 G
CF.HDR= 000020        C.BUF2  000024        FS.EXI= 001000        LF.PAC= 000200        LS.OPT= 000400 G
CF.LB = 100000        C.CNT   000020        FS.GET= 006000        LF.RDY= 040000        LS.PRO= 000002 G
CF.LIN= 000002        C.CNT1  000020        FS.HLT= 000000        LF.REA= 010000        LS.PWF= 040000 G
CF.LOG= 000020        C.CNT2  000030        FS.INI= 000000        LF.SER= 000040        LS.TOP= 002000 G
CF.MDM= 000001        C.FLG   000022        FS.KIL= 000000        LF.TIM= 000010        LS.UNF= 020000 G
CF.SOM= 000010        C.FLG1  000022        FS.LCL= 100000        LF.UNL= 020000        LS.X25= 000012 G
```

```
                                                  F 15
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:02  Page 17-2
Symbol table
```

```
          .TITLE  NTITIM
          .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       NTINIT - START COMM EXEC TIMERS
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
                                        .SBTTL  XLT16 - DEALLOCATE LINE TABLE
                                ;+
                                ; XLT16 - DE-ALLOCATE A 16 BIT ALLOCATION LINE TABLE
                                ;
                                ; INPUTS:
                                ;       R3=LINE TABLE ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       R0,R1,R2,R3=DESTROYED
                                ;-
005452                          XLT16:  CALL    FLT16           ; FIND 16 BIT LINE TABLE
005456  016703  000010'         XLT:    MOV     UBA,R3          ; WAS UNLOAD ENTRY FOUND?
005462  001001                          BNE     120$            ; IF NE, YES
005464                                  CRASH                   ; LINE TABLE NOT FOUND IN UNLOAD DATA!!!
005466  010367  000112'         120$:   MOV     R3,UPTR         ; SAVE POINTER TO IT
005472  005023                          CLR     (R3)+           ; SKIP THE PDV INDEX
005474  011300                          MOV     (R3),R0         ; GET LINE TABLE ADDRESS
005476  005023                          CLR     (R3)+           ; ZERO IT (CLEARS C-BIT)
005500  011301                          MOV     (R3),R1         ; GET LINE TABLE SIZE
005502  042701  000003                  BIC     #3,R1           ; CLEAR EXTRANEOUS BITS
005506                                  CALL    .DEA16          ; DE-ALLOCATE LINE TABLE
005512  132767  000001  000200'         BITB    #PR.LTA,PRLTA   ; PARTITION LINE TABLE?
005520  001411                          BEQ     140$            ; NO IF EQUAL
005522  142767  000001  000200'         BICB    #PR.LTA,PRLTA   ; REASONABLE HERE (I THINK!!)
005530  016701  000176'                 MOV     ICBSZ,R1        ; SIZE OF ICB
005534  016700  000174'                 MOV     ICBSAV,R0       ; ..AND IT'S ADDRESS

                                        .IF DF  R$$MPL                                  ;[MP01]

                                        TST     .MPRO           ; MULTIPROCESSOR SYSTEM ?       ;[MP01]
                                        BLT     130$            ; IF IF NO - BRANCH             ;[MP01]
                                        CALL    DEICP           ; IF YES DEALLOCATE - ICB'S     ;[MP01]
                                        BR      140$            ; CONTINUE                      ;[MP01]
                                130$:   BIT     #DF.DAS,$DFLAG  ; D-SPACE SYSTEM ?              ;[MP01]
                                        BEQ     135$            ; IF NO - BRANCH                ;[MP01]
                                        CALL    $DEAIC          ; DEALLOCATE ICB'S              ;[MP01]
                                        BR      140$            ; CONTINUE                      ;[MP01]
                                                                                                ;[MP01]
                                135$:
                                        ADD     #4,R1           ; ADJUST FOR LINK WORD AND FIELD ;[MP01]
                                        SUB     #4,R0           ;   ...                         ;[MP01]
                                        .IFTF
005540                                  CALL    .DEA16          ; DO DEALLOCATE

                                        .ENDC                                                   ;[MP01]

005544  011346                  140$:   MOV     (R3),-(SP)      ; SAVE LINE TABLE SIZE
005546  005023                          CLR     (R3)+           ; ZERO IT
005550  006026                          ROR     (SP)+           ; ANY SUB-ALLOCATIONS?
005552  103047                          BCC     200$            ; IF CC, NO
005554  005046                          CLR     -(SP)           ; CALCULATE ALLOCATION COUNT
005556  005046                          CLR     -(SP)           ; AND LIBRARY COUNT
005560  152316                          BISB    (R3)+,(SP)      ;       ...
005562  152366  000002                  BISB    (R3)+,2(SP)     ;       ...
005566  005063  177776                  CLR     -2(R3)          ; ZERO THE COUNTS
005572  012667  000116'                 MOV     (SP)+,COUNT     ; GET NUMBER OF ALLOCATIONS
005576  001421                          BEQ     180$            ; IF ZERO, NO WORK TO DO
```

F 2

```
VUNL       CREATED BY  MACRO  ON 16-SEP-85 AT 07:31     PAGE 6
SYMBOL  CROSS REFERENCE                                 CREF   04.00

SYMBOL  VALUE              REFERENCES

$PDVNM  = ******  GX        9-412      21-1301
$PDVTA  = ******  GX        9-413      9-420      14-793     21-1300
$PNAME    000000 RG        #6-111
$PRAVL  = ******  GX       *24-1816   *24-1817    24-1823
$PUTAD  = ******  GX        20-1226    23-1756    23-1776
$PUTRC  = ******  GX        17-1050    21-1271    21-1394    21-1484
$RADDR  = ******  GX       *7-208     *8-322     *8-334      10-442    *10-453    *10-466    *10-479    *10-485    *10-497
                           *11-524    *14-700    *15-827    *15-836    *17-962    *17-1002   *17-1046   *17-1050   *18-1116
                            21-1263   *21-1264   *21-1271   *21-1323   *21-1394   *21-1467   *21-1496   *21-1509   *21-1517
                           *21-1538   *21-1547   *21-1567    23-1740   *23-1745   *23-1775   *24-1814   *25-1849   *25-1853
$RBLCK  = ******  GX       *21-1502   *21-1531   *21-1555   *21-1574   *23-1742    23-1744   *23-1774
$RDBLH  = ******  GX       *21-1334   *21-1335
$RSIZE  = ******  GX       *18-1090   *21-1312   *22-1595   *23-1739
$SAVAL  = ******  GX        10-441     20-1204
$SDBLH  = ******  GX       *21-1336   *21-1337
$SLTA   = ******  GX        12-559     14-649     20-1211   *21-1292
$SLTMA  = ******  GX        15-860     19-1174    21-1286
$SLTNM  = ******  GX        15-861     19-1175    21-1287
$UNBLK  = ******  GX        20-1209    20-1225
$UNBSZ  = ******  GX        23-1732
$VFLAG  = ******  GX        8-314
$VUNB   = ******  GX        20-1213
$XLBR   = ******  GX        23-1721
$XLIN     001320 RG        #12-556
$XLINK  = ******  GX        7-231
$XPAR   = ******  GX        8-349      8-353      21-1367    21-1373    21-1380
$XPRO     000132 RG        #8-260
$XPSI   = ******  GX        21-1350    21-1352
$XSNA   = ******  GX        21-1353    21-1355
$$      = 177777            13-627     13-627    #13-627    #13-628     17-1066    17-1066   #17-1066    17-1067    17-1067
                           #17-1067   #17-1068    20-1235    20-1235   #20-1235   #20-1236
$$$     = 003600 R         #13-627     13-628    #17-1066    17-1068   #20-1235    20-1236
$$$$    = 003664 R         #13-628    #17-1068   #20-1236
.CXALL  = ******  GX        21-1359
.CXCSR  = ******  GX        20-1205
.CXLEN  = ******  GX        21-1360
.CXPCB  = ******  GX        21-1379
.CXUNL  = ******  GX        22-1598    23-1734   *23-1736    23-1741    23-1746
.CXVEC  = ******  GX        7-203      7-224
.DEAPR  = ******  GX        24-1819
.DEA16    006144 R          23-1664    23-1686    23-1707   #24-1804
.NETPF  = ******  GX        21-1394
.NS0    = ******  GX        6-166
.NS1    = ******  GX        6-167
.NS2    = ******  GX        6-168
.NS3    = ******  GX        6-169
.NS4    = ******  GX        6-170
.NS5    = ******  GX        6-171
.NS6    = ******  GX        6-172
.NS7    = ******  GX        6-173
.PLGTH  = ******  GX        8-322      8-334      24-1814
.SBPCB  = ******  GX        21-1371
```

G 2

```
VUNL       CREATED BY  MACRO  ON 16-SEP-85 AT 07:31     PAGE 7
SYMBOL  CROSS REFERENCE                                 CREF   04.00
```

```
    249                                                 .SBTTL  DEADST - Deallocate destination descriptors                  ;[TD001]
    250                                 ;+                                                                                   ;[TD001]
    251                                 ;                                                                                    ;[TD001]
    252                                 ; *** DEADST - Deallocate destination descriptor blocks                              ;[TD001]
    253                                 ;                                                                                    ;[TD001]
    254                                 ; INPUTS:                                                                            ;[TD001]
    255                                 ;       R3 - Address of listhead                                                     ;[TD001]
    256                                 ;                                                                                    ;[TD001]
    257                                 ; OUTPUTS:                                                                           ;[TD001]
    258                                 ;       The destination descriptor blocks are deallocated                            ;[TD001]
    259                                 ;                                                                                    ;[TD001]
    260                                 ;-                                                                                   ;[TD001]
    261 000552                          DEADST: CALL    $SAVAL          ; Save all registers                                 ;[TD001]
    262 000556                                  SAVMAP                  ; Save mapping                                       ;[TD001]
    263 000562                                  GETRV   R3,#2           ; Read in listhead                                   ;[TD001]
    264 000600  016746  000000G                 MOV     $BFR,-(SP)      ; Stack address of first block                       ;[TD001]
    265 000604  005067  000000G                 CLR     $BFR            ; and zero listhead                                  ;[TD001]
    266 000610                                  PUTRC                   ; Write updated listhead                             ;[TD001]
    267                                                                                                                      ;[TD001]
    268 000614  012600                  10$:    MOV     (SP)+,R0        ; Get unmapped address of next block                 ;[TD001]
    269 000616  001462                          BEQ     50$             ; If EQ, end of list                                 ;[TD001]
    270 000620                                  CEACC$  R0,R2           ; Convert to mapped address                          ;[TD001]
    271 000630  032767  000000G 000000G         BIT     #PF.XDF,$PFLAG  ; Check for extended data format                     ;[TD001]
    272 000636  001423                          BEQ     30$             ; Branch if not extended format                      ;[TD001]
    273                                 ;                                                                                    ;[TD001]
    274                                 ; Compute total length of extended fields                                            ;[TD001]
    275                                 ;                                                                                    ;[TD001]
    276 000640                                  GETAD   R2,#D$GLEN      ; Read fixed part of block                           ;[TD001]
    277 000656  016746  000000G                 MOV     $BFR,-(SP)      ; Save link to next block                            ;[TD001]
    278 000662  012701  000032                  MOV     #D$GLEN,R1      ; Get fixed length of block                          ;[TD001]
    279 000666  012702  000022G                 MOV     #$BFR+D$DSL,R2  ; Address first length field                         ;[TD001]
    280 000672  112203                  20$:    MOVB    (R2)+,R3        ; Read length field                                  ;[TD001]
    281 000674  060301                          ADD     R3,R1           ; Add to total block length                          ;[TD001]
    282 000676  020227  000027G                 CMP     R2,#$BFR+D$ACL  ; Check if last length field added                   ;[TD001]
    283 000702  101773                          BLOS    20$                                                                  ;[TD001]
    284 000704  000413                          BR      40$             ; Check remaining fields                             ;[TD001]
    285                                 ;                                                                                    ;[TD001]
    286                                 ; Compute length of old-format fields                                                ;[TD001]
    287                                 ;                                                                                    ;[TD001]
    288 000706                          30$:    GETAD   R2,#D$FLEN      ; Read fixed part of block                           ;[TD001]
    289 000724  016746  000000G                 MOV     $BFR,-(SP)      ; Save link to next block                            ;[TD001]
    290 000730  012701  000032                  MOV     #D$FLEN,R1      ; Get fixed length of block                          ;[TD001]
    291                                 ;                                                                                    ;[TD001]
    292                                 ; Add length of call mask/value and DTE fields                                       ;[TD001]
    293                                 ;                                                                                    ;[TD001]
    294 000734  116703  000030G         40$:    MOVB    $BFR+D$DATL,R3  ; Get length of call mask/data                       ;[TD001]
    295 000740  006303                          ASL     R3              ; Double, since two fields of same length            ;[TD001]
    296 000742  060301                          ADD     R3,R1           ; Increase length to deallocate                      ;[TD001]
    297 000744  116703  000031G                 MOVB    $BFR+D$DTEL,R3  ; Get number of digits in DTE number                 ;[TD001]
    298 000750  006203                          ASR     R3              ; Compute bytes occupied by DTE number               ;[TD001]
    299 000752  005503                          ADC     R3                                                                   ;[TD001]
    300 000754  060301                          ADD     R3,R1           ; Increase length to deallocate                      ;[TD001]
    301                                                                                                                      ;[TD001]
    302 000756                                  CALL    $DEPOL          ; Deallocate destination                             ;[TD001]
    303 000762  000714                          BR      10$             ; and go again                                       ;[TD001]
    304                                                                                                                      ;[TD001]
    305 000764                          50$:    RESMAP                  ; Restore mapping                                    ;[TD001]
```

```
VUNPSI     CREATED BY  MACRO  ON 17-JUL-85 AT 12:15      PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

ASL$             #5-58     12-356    15-475
CALL              8-119     8-120     8-123     8-126     8-128     8-130     8-135     8-136     8-137     8-138
                  8-139     8-140     8-153     9-170     9-179     9-185     9-188     9-192     9-193     9-197
                 10-216    10-225    10-231    10-234    10-238    10-239    10-243    11-261    11-263    11-266
                 11-270    11-276    11-288    11-302    12-324    12-330    12-334    12-335    12-347    12-350
                 12-361    12-365    13-385    13-388    13-391    13-392    13-395    13-408    13-416    14-436
                 14-440    14-442    15-460    15-464    15-468    15-469    15-487    16-504    16-507    16-511
                 16-514    16-518    17-537    17-539    17-542    17-546    17-547    17-551
CEACC$           #5-59      9-192    10-238    11-270    12-334    13-392    17-546
CHNDF$           #5-58      5-62
CUGDF$           #5-57      5-63
DSTDF$           #5-57      5-64
DTEDF$           #5-57      5-65
GETAD            #5-59      9-193    10-239    11-276    11-288    12-335    13-395    17-547
GETRV            #5-59      9-185    10-231    11-263    12-330    12-347    13-385    13-388    14-436    15-464
                 15-469    16-507    16-514    17-539
PHBDF$           #5-57      5-66
PUTAD            #5-59
PUTRC            #5-59      9-188    10-234    11-266    12-365    14-440    15-468    17-542
PVCDF$           #5-57      5-67
RAND             #9-185     9-185    #9-188     9-188    #9-193     9-193   #10-231    10-231   #10-234    10-234
                #10-239    10-239   #11-263    11-263   #11-266    11-266   #11-276    11-276   #11-288    11-288
                #12-330    12-330   #12-335    12-335   #12-347    12-347   #12-365    12-365   #13-385    13-385
                #13-388    13-388   #13-395    13-395   #14-436    14-436   #14-440    14-440   #15-464    15-464
                #15-468    15-468   #15-469    15-469   #16-507    16-507   #16-514    16-514   #17-539    17-539
                #17-542    17-542   #17-547    17-547
RDTDF$           #5-57      5-68
RESMAP           #5-82      9-200    10-246    11-305    12-366    13-410    17-555
RESRG            #5-58     12-359    12-360    13-409    15-485    15-486    17-552
RETURN            8-156     9-201    10-247    11-306    12-367    13-417    14-443    15-488    16-520    17-556
RNWDF$           #5-57      5-74
SAVMAP           #5-78      9-184    10-230    11-262    12-325    13-406    17-538
SAVRG            #5-58     12-338    12-343    13-407    15-471    15-472    17-550
SVCDF$           #5-58      5-69
SWTCHI           #5-59     12-327    12-331
SWTCHO           #5-59
TCBDF$           #5-58      5-72
TRADF$           #5-58      5-73
VNPDBG           #5-60      8-116
XCBDF$           #5-57      5-70
X29DF$           #5-58      5-71
```

```
**FILE**ID**VVCT
```

```
VVCT       CREATED BY  MACRO  ON 29-JUN-85 AT 03:12     PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME        REFERENCES

ASR$              #5-57
CALL               6-120      7-254      8-280
CALLR             #5-58
EMSG$             #5-57
EMSG$R            #5-57
GETAD             #5-57
GETRV             #5-57
NTLER$            #5-57
PUTAD             #5-57
PUTRC             #5-57       6-120
RAND              #6-120      6-120
RESRG             #5-58       6-130
RETURN             6-131      7-255      8-282      9-331
SAVRG             #5-58       6-86
SWTCHI            #5-58
SWTCHO            #5-57       6-117      6-129
```

```
**FILE**ID**VXIT
```

```
    115 000166                        60$:    RETURN
    116
    117
```

```
**FILE**ID**NTICEX




NN      NN  TTTTTTTTTT    IIIIII      CCCCCCCC  EEEEEEEEEE  XX      XX
NN      NN  TTTTTTTTTT    IIIIII      CCCCCCCC  EEEEEEEEEE  XX      XX
NN      NN      TT          II      CC          EE          XX      XX
NN      NN      TT          II      CC          EE          XX      XX
NNNN    NN      TT          II      CC          EE            XX  XX
NNNN    NN      TT          II      CC          EE            XX  XX
NN  NN  NN      TT          II      CC          EEEEEEEE        XX
NN  NN  NN      TT          II      CC          EEEEEEEE        XX
NN    NNNN      TT          II      CC          EE            XX  XX
NN    NNNN      TT          II      CC          EE            XX  XX
NN      NN      TT          II      CC          EE          XX      XX          ....
NN      NN      TT          II      CC          EE          XX      XX          ....
NN      NN      TT        IIIIII      CCCCCCCC  EEEEEEEEEE  XX      XX          ....
NN      NN      TT        IIIIII      CCCCCCCC  EEEEEEEEEE  XX      XX          ....



LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

```
NTICEX  MACRO V05.03b Saturday 29-Jun-85 03:25
Table of contents
```

```
                                                                    F  8
NTICEX     CREATED BY  MACRO  ON 29-JUN-85 AT 03:25     PAGE 3
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL   VALUE               REFERENCES

TPS      = 177564            #5-74
UBMPR    = 170200            #5-74
UISAR0   = 177640            #5-74
UISAR1   = 177642            #5-74
UMRPT    = ******   GX         7-246
X$$MCB   = ******              5-75        5-75
ZF.COU   = 001000            #5-75
ZF.DDM   = 000001            #5-75
ZF.DIA   = 004000            #5-75
ZF.DLC   = 000002            #5-75
ZF.DVP   = 100000            #5-75
ZF.INI   = 040000            #5-75         7-193
ZF.KMX   = 000020            #5-75
ZF.LLC   = 000004            #5-75
ZF.LMC   = 000100            #5-75
ZF.MAN   = 020000            #5-75
ZF.MFL   = 000010            #5-75
ZF.MTM   = 000400            #5-75
ZF.MUX   = 000040            #5-75
ZF.PSE   = 002000            #5-75
ZF.SLI   = 010000            #5-75
ZF.TIM   = 000200            #5-75
ZF.X3P   = 000000            #5-75
ZS.ASN   = 100000            #5-75
ZS.BSY   = 140000            #5-75
Z.AVL      000014            #5-75
Z.DAT      000016            #5-75         7-197
Z.DSP      000000            #5-75         5-75
Z.FLG      000010            #5-75         7-193
Z.LEN    = 000016            #5-75
Z.LLN      000006            #5-75
Z.MAP      000020            #5-75
Z.NAM      000004            #5-75         7-203
Z.PCB      000012            #5-75         7-195
Z.SCH      000007            #5-75
$ERR10     000000 R          #5-90         6-156
$NICEX     000000 RG         #6-135
$NICSR   = ******   GX         7-268
$NIERR   = ******   GX         5-90
$NIFLG   = ******   GX        *6-147
$NIMFL   = ******   GX         6-149
$NINIL   = ******   GX         6-140       6-155
$NITIM   = ******   GX         6-151       7-279
$NIUMR   = ******   GX         7-250
$NTLFC   = ******   GX        *6-139       6-142       6-145      *6-154
.BASEB   = 140000            #5-74         7-231       7-234
.CXCSR   = ******   GX         7-234
```

```
                                                                    G  8
NTICEX     CREATED BY  MACRO  ON 29-JUN-85 AT 03:25     PAGE 4
```

```
**FILE**ID**NTILON
```

```
NN      NN   TTTTTTTTTT     IIIIII     LL             OOOOOO     NN      NN
NN      NN   TTTTTTTTTT     IIIIII     LL             OOOOOO     NN      NN
NN      NN       TT           II       LL           OO      OO   NN      NN
NN      NN       TT           II       LL           OO      OO   NN      NN
NNNN    NN       TT           II       LL           OO      OO   NNNN    NN
NNNN    NN       TT           II       LL           OO      OO   NNNN    NN
NN  NN  NN       TT           II       LL           OO      OO   NN  NN  NN
NN  NN  NN       TT           II       LL           OO      OO   NN  NN  NN
NN    NNNN       TT           II       LL           OO      OO   NN    NNNN
NN    NNNN       TT           II       LL           OO      OO   NN    NNNN
NN      NN       TT           II       LL           OO      OO   NN      NN      ....
NN      NN       TT           II       LL           OO      OO   NN      NN      ....
NN      NN       TT         IIIIII     LLLLLLLLLL     OOOOOO     NN      NN      ....
NN      NN       TT         IIIIII     LLLLLLLLLL     OOOOOO     NN      NN      ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                                                                 F 10
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:26  Page 9-3
Symbol table

S4.ANI= 000400            T$NVR   000001            UU.SIO= 001000            U.TTYP  000046            W.BLGH  000020
S4.AVO= 001000            T$RPRI  000040            UU.SPC= 000040            U.TUX   000024            W.BLPD  000016
S4.BLK= 002000            T$SVC   000034            U.ACB   000062            U.UHVR= 000107            W.BLVR  000002
S4.DEC= 004000            T$T5    000030            U.ACP = 000032            U.UIC   000044            W.BNPD  000015
S4.DLO= 000100            T$T6    000032            U.ADMA  000066            U.UNFL= 000052            W.BOFF  000012
S4.EDT= 010000            T$$KMG= 000000            U.AFLG  000064            U.UNIT  000006            W.BPCB  000000
S4.HFF= 000020            T$$MIN= 000000            U.ATT   000022            U.UNSZ= 000114            W.BSIZ  000010
S4.HFL= 000007            UBMPR = 170200            U.BPKT= 000044            U.UNTI= 000060            X$$DBT= 000000
S4.HHT= 000040            UC.ALG= 000200            U.BUF   000024            U.USVR= 000106            ZF.COU= 001000
S4.HSY= 000200            UC.ATT= 000010            U.CBF = 000032            U.UTIL= 000036            ZF.DDM= 000001
S4.RGS= 020000            UC.KIL= 000004            U.CLI   177772            U.VCB = 000034            ZF.DIA= 004000
S4.SFC= 040000            UC.LGH= 000003            U.CNT   000030            U.VSER= 000120            ZF.DLC= 000002
S4.VFL= 000010            UC.NPR= 000100            U.COTQ  000030            U.2MED= 000070            ZF.DVP= 100000
S5.ABP= 100000            UC.PWF= 000020            U.CTCB  000026            U2.AT.= 000020            ZF.INI= 040000
S5.BCC= 020000            UC.QUE= 000040            U.CTL   000004            U2.CRT= 002000            ZF.KMX= 000020
S5.DAO= 040000            UD.UNS= 000000            U.CTYP  000052            U2.DH1= 100000            ZF.LLC= 000004
S5.HPC= 000014            UD.160= 000004            U.CW1   000010            U2.DJ1= 040000            ZF.LMC= 000100
S5.HPO= 000020            UD.200= 000001            U.CW2   000012            U2.DZ1= 000100            ZF.MAN= 020000
S5.ITI= 000100            UD.556= 000002            U.CW3   000014            U2.ESC= 001000            ZF.MFL= 000010
S5.OXF= 000040            UD.625= 000005            U.CW4   000016            U2.HFF= 010000            ZF.MTM= 000400
S5.RPO= 002000            UD.8K = 000006            U.CYL = 000104            U2.HLD= 000040            ZF.MUX= 000040
S5.SW1= 000001            UD.800= 000003            U.DCB   000000            U2.LOG= 000400            ZF.PSE= 002000
S5.TMM= 000002            UISAR0= 177640            U.FCDE= 000042            U2.LWC= 000001            ZF.SLI= 010000
S5.VER= 010000            UISAR1= 177642            U.FNUM= 000040            U2.L3S= 000004            ZF.TIM= 000200
S5.XOF= 000004            UM.CLI= 000036            U.GRP = 000102            U2.L8S= 010000            ZF.X3P= 000000
S5.XON= 000010            UM.DSB= 000200            U.KCSR= 000032            U2.NEC= 004000            ZS.ASN= 100000
S6.EIO= 000400            UM.NBR= 000400            U.KCS6= 000034            U2.PRV= 000010            ZS.BSY= 140000
S6.RDI= 100000            UM.OVR= 000001            U.LUIC  177774            U2.RMT= 020000            Z.AVL   000014
S6.RLU= 001000            US.ABO= 000001            U.MEDI= 000040            U2.R04= 100000            Z.DAT   000016
TPS   = 177564            US.BSP= 000002            U.MLUN= 000050            U2.SCS= 000004            Z.DSP   000000
T$FLAG  000044            US.BSY= 000200            U.MUP   177772            U2.SLV= 000200            Z.FLG   000010
T$LIF   000013            US.FOR= 000040            U.OWN   177776            U2.VT5= 000002            Z.LEN = 000016
T$LIFL  000013            US.FRK= 000002            U.RBNS= 000112            U2.7CH= 010000            Z.LLN   000006
T$LIFO  000013            US.KPF= 000001            U.RCTC= 000113            U3.OPA= 100000            Z.MAP   000020
T$LIFS  000013            US.LAB= 000004            U.RCTS= 000110            U3.PAR= 040000            Z.NAM   000004
T$LIN   000000            US.MDE= 000002            U.RED   000002            U3.UPC= 020000            Z.PCB   000012
T$LIPS  000006            US.MDM= 000020            U.RED2  000034            U4.CR = 000100            Z.SCH   000007
T$LLD   0000 2            US.MNT= 000100            U.SCB   000020            VF$RCV= 100000            $DBECL= ****** GX
T$LLDC  000045            US.OFL= 000001            U.SHST= 000076            VF$XMT= 040000            $DBXPT= ****** GX
T$LLDL  000012            US.PUB= 000004            U.SHUN= 000074            V$FLG   000000            $ERR11  000052RG      002
T$LLDO  000012            US.PWF= 000010            U.SPC = 000036            V$LEN   000022            $ERR15  000142RG      002
T$LLDS  000012            US.RED= 000002            U.STS   000005            V$RCV   000002            $ERR17  000202RG      002
T$LLEN  000046            US.SHR= 000001            U.ST2   000007            V$XMT   000012            $ERR18  000256R       002
T$LOPR  000002            US.SPU= 000002            U.SUB = 000036            V$$CTR= 001000            $ERR19  000316RG      002
T$LTCL  000024            US.UMD= 000010            U.TCHP  000042            V.CULT  000004            $HEADR= ****** GX
T$LTIM  000026            US.VV = 000001            U.TCVP  000043            V.GENQ  000016            $MUL  = ****** GX
T$LTPR  000014            US.WCK= 000010            U.TFLK  000040            V.MBXQ  000012            $NIERR= ****** GX
T$LTPS  000020            UU.ABO= 000400            U.TFRQ  000036            V.TIME  000003            $NILON  000000RG
T$NAPL  000004            UU.ATN= 000100            U.TIXL  000060            V.TIMR  000002            $NIMFE= ****** GX
T$NFE   000000            UU.AVN= 000004            U.TLPP  000053            V.TRCT  000000            $NIXPT= ****** GX
T$NLEN  000010            UU.GUS= 000010            U.TMTI  000047            V.ULT   000006            $$$OST= 000010
T$NNUL  000002            UU.IOS= 002000            U.TRCK= 000100            V.ULTE  000010            $$$T1 = 000002
T$NOPL  000006            UU.ONL= 000020            U.TSTA  000026            V.VBL = 000022            $$$T2 = 000046R       002
T$NRNI  000042            UU.RCT= 000002            U.TST5  000054            W.BATT  000006            .BASEB= 140000
T$NRPL  000005            UU.RDY= 000200            U.TST6  000056            W.BFPD  000014            .$$$$.= 000034
T$NRUL  000007            UU.SER= 000001            U.TTAB  000050            W.BHVR  000004
```

```
                                                                                 G 10
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:26  Page 9-4
Symbol table
```

```
000424  000461                              BR      60$                     ;   AND RETURN TO COMMON CODE

000426  032765  002000  000000      50$:    BIT     #LF.ENA,L.FLG(R5)       ;SHOULD THIS LINE BE ENABLED?
000434  001455                              BEQ     60$                     ;NO
000436  011246                              MOV     (R2),-(SP)              ;GET REVERSE MAPPING VALUE
000440  000316                              SWAB    (SP)                    ;CALCULATE OWNER LLC INDEX
000442  005001                              CLR     R1                      ; ...
000444  111601                              MOVB    (SP),R1                 ; ...
000446  005726                              TST     (SP)+                   ;CLEAN UP STACK
000450  067701  000000G                     ADD     @PDVTA,R1               ;GET LLC PDV ADDRESS
000454  011101                              MOV     (R1),R1                 ; ...
000456  022761  114224  000004              CMP     #^RXPT,Z.NAM(R1)        ;XPT ?
000464  001030                              BNE     56$                     ;IF NO - BRANCH

000466  016701  000000G                     MOV     $DBECL,R1               ;COPY ECL DATA BASE ADDRESS
000472  132761  000040  000005              BITB    #NF$MOU,N$FLG(R1)       ;IS NETWORK ON ?
000500  001001                              BNE     51$                     ;IF YES - BRANCH
000502  000432                              BR      60$                     ;IF NO - DO NOT ENABLE CIRCUIT

000504  017701  000000G             51$:    MOV     @DECPT,R1               ;GET DECNET HOME BLOCK ADDRESS
000510  005761  000060                      TST     D$NN(R1)                ;ROUTING NODE ?
000514  001014                              BNE     56$                     ;IF YES - BRANCH
000516  005767  177262                      TST     ACTCIR                  ;OTHER LINES ENABLED ?
000522  001407                              BEQ     55$                     ;IF NO - BRANCH
000524  042765  002000  000000              BIC     #LF.ENA,L.FLG(R5)       ;DISABLE LINE
000532                                      EMSG$   12                      ;PRINT MSG
000540  000413                              BR      60$                     ;CONTINUE

000542  005267  177236              55$:    INC     ACTCIR                  ;COUNT ANOTHER LINE ENABLED
000546                              56$:    CALL    ENALIN                  ;ENABLE THE LINE
000552  103406                              BCS     60$                     ;BR IF SPECIFIED LLC DOES NOT OWN LINE
000554  052765  100000  000000              BIS     #LF.ACT,L.FLG(R5)       ;SET LINE ACTIVE
000562  112765  000000  000020              MOVB    #LN.ON,L.NMST(R5)       ;SET MANAGEMENT STATE TO 'ON'

000570  062702  000002              60$:    ADD     #2,R2                   ;NEXT SYSTEM LINE NUMBER
000574  005316                              DEC     (SP)                    ;REDUCE SYSTEM LINE COUNT
000576  001402                              BEQ     65$                     ;ANY MORE TO SCAN?
000600                                      CALLR   ENAC                    ;IF YES - CHECK NEXT CIRCUIT

000604  005726                      65$:    TST     (SP)+                   ;NO ... CLEAN UP THE STACK

000606                              70$:    RETURN
```

Table of contents

```
     5-   54    MACRO DEFINITIONS
     6-   69    LOCAL DATA
     7-   89    MARKED FOR LOAD SCAN
     9-  172    DLMMFL - DLM MARKED FOR LOAD SCAN
```

```
NTIMFL     CREATED BY  MACRO  ON 29-JUN-85 AT 03:27     PAGE 5
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

CALL              7-122       8-141       8-152       8-165       9-201
EMSG$            #5-58        8-165
EMSG$R           #5-58        7-127
NHWDF$           #5-58        5-66
NTLDF$           #5-58        5-67
NTLER$           #5-58        6-80        6-81
PDVDF$           #5-58        5-64
RESRG            #5-59        8-166
RETURN            8-170       9-208
SAVRG            #5-59        8-164
SLTDF$           #5-58        5-65
SOB               7-129
```

```
**FILE**ID**NTINIT
```

```
000364                          ENAR$S                          ;ENABLE AST'S AGAIN
000372                          CALLR   NTIGRQ                  ;PROCESS NEXT REQUEST
                                .IFF    ;NDF R$$11S

                                  .IF DF S$$BAS
                                DSAR$S                          ;DISABLE AST RECOGNITION
                                TST     $NIAST                  ;NOTIFY USERS VIA AST?
                                BEQ     110$                    ;BR IF NO
                                CALL    PUTAST                  ;ELSE NOTIFY USER VIA AST
                                BR      120$                    ;AND CONTINUE

                        110$:   DIR$    #NTISEN                 ;SEND COMPLETION TO THE REQUESTOR
                        120$:   ENAR$S                          ;ENABLE AST'S AGAIN
                                  .ENDC

                                EXIT$S

                                .ENDC

```

```
Symbol table

LS.11D= 004000 G          NTLPT   000652RG    002 QASTT   000632RG    002 TKPS    000636RG    002 T2.STP= 000020
LX.CEX= 000004 G          N$$ACC= 000001          Q$$OPT= 000010              TKTCB   000630RG    002 T2.WFR= 000001
LX.LIN= 000006 G          N$$BUF= 000001          R$$DER= 000000              TK100   000662RG    002 T3.ACP= 100000
LX.PRO= 000005 G          N$$LDV= 000001          R$$K11= 000001              TPS   = 177564                T3.CAL= 000100
L$$ASG= 000000            N$$MCP= 000001          R$$SND= 000000              TS.BLK= 171700                T3.CLI= 001000
L$$DRV= 000000            N$$MLL= 000001          R$$11M= 000000              TS.CKP= 000200                T3.GFL= 000010
L$$P11= 000001            N$$MOV= 000010          R.QSGC= 000015              TS.CKR= 000100                T3.MCR= 004000
L$$11R= 000000            N$$NCT= 000001          R.QSPC= 000014              TS.EXE= 100000                T3.NET= 000020
L.COST  000015            N$$PEM= 000001          R.QSPN= 000006              TS.HLD= 002000                T3.NSD= 000200
L.CTL   000012            PDVID   000704RG    002 R.QSPR= 000012              TS.MSG= 020000                T3.PMD= 040000
L.CVA   177776            PDVNM   000702RG    002 R.QSTN= 000002              TS.NRP= 010000                T3.PRV= 010000
L.DDM   000002            PDVTA   000712RG    002 R.VDBA= 000006              TS.OUT= 000400                T3.REM= 020000
L.DDS   000004            PIRQ  = 177772          R.VDTN= 000002              TS.RDN= 040000                T3.ROV= 000040
L.DLC   000003            PMODE = 030000          R.VXBA= 000006              TS.RUN= 004000                T3.RST= 000400
L.DLM   000006            PR0   = 000000          R.VXTN= 000002              TS.STP= 001000                T3.SLV= 002000
L.DLS   000010            PR1   = 000040          SF.ACT= 000200              T$$KMG= 000000                T3.SWS= 000002
L.FLG   000000            PR2   = 000100          SF.ENA= 000100              T$$MIN= 000000                UBMPR = 170200
L.KRBA  000016            PR3   = 000140          SF.LPB= 000004              T.ACTL  000052                UISAR0= 177640
L.LEN = 000022            PR4   = 000200          SF.MFL= 000040              T.ASTL  000016                UISAR1= 177642
L.MPF   000022            PR5   = 000240          SF.PAC= 000020              T.ATT   000062                UMRPT   000614RG    002
L.NMST  000020            PR6   = 000300          SF.REA= 000010              T.CPCB  000004                VECLN = 000043 G
L.NSTA  000014            PR7   = 000340          SF.SER= 000001              T.DPRI  000040                VF.LIB= 000001
L.OWNR  000021            PS    = 177776          SF.SVC= 000002              T.EFLG  000022                V$$CTR= 001000
L.UNT   000013            PSIPT   000670RG    002 SF.UNL= 000040              T.IOC   000003                V.ADD   000006
MPAR  = 172100            PS.APR= 000007          SLTMA   000646RG    002 T.LBN   000041                V.BIAS  000004
MPCSR = 177746            PS.CHK= 010000          SLTNM   000650RG    002 T.LDV   000044                V.FLG   000002
MRKTEF= 000001            PS.CKP= 040000          SPARE = 000010              T.LNK   000000                V.LEN   000012
M$$CRB= 000124            PS.CKR= 020000          SP.EIP= 000001              T.MXSZ  000050                V.LNK   000000
M$$CRX= 000000            PS.COM= 000200          SP.ENB= 000002              T.NAM   000006                V.PDV   000004
M$$FCS= 000000            PS.DEL= 000010          SP.LOG= 000004              T.OFF   000066                V.SIZ   000010
M$$MGE= 000000            PS.DRV= 000020          SRSTD   000616RG    002 T.PCB   000046                X$$DBT= 000000
M$$NET= 000000            PS.FXD= 004000          SWR   = 177570              T.PRI   000002                ZF.COU= 001000
M$$OVR= 000000            PS.LIO= 001000          S$$WRG= 000000              T.RCVL  000012                ZF.DDM= 000001
M.BFVH  000011            PS.NSF= 000400          S$$YSZ= 007600              T.RRFL  000072                ZF.DIA= 004000
M.BFVL  000012            PS.OUT= 100000          S.BMSK  177776              T.SAST  000054                ZF.DLC= 000002
M.KTAE= 000010            PS.PER= 002000          S.BMSV  177774              T.SRCT  000071                ZF.DVP= 100000
M.KTEF= 000002            PS.PIC= 000100          S.CON   000010              T.STAT  000032                ZF.INI= 040000
M.KTMG= 000004            PS.SYS= 000040          S.COST  000001              T.ST2   000034                ZF.KMX= 000020
M.KTUN= 000006            PUMR    000624RG    002 S.CSR   000012              T.ST3   000036                ZF.LLC= 000004
M.LGTH= 000014            PUTAST  001222R         S.CTM   000006              T.TCBL  000030                ZF.LMC= 000100
M.LNK   000000            P$$P45= 000000          S.DABA= 000006              T.TIO   000057                ZF.MAN= 020000
M.UMRA  000002            P$$WRD= 000000          S.DAEF= 000010              T.TKSZ  000060                ZF.MFL= 000010
M.UMRN  000004            P.BLKS  000016          S.DATN= 000002              T.UCB   000026                ZF.MTM= 000400
M.UMVH  000010            P.BUSY  000024          S.DMCS  000020              T1SCL   000674RG    002 ZF.MUX= 000040
M.UMVL  000006            P.IOC   000003          S.FLG   000000              T100C   000656RG    002 ZF.PSE= 002000
NTIBSC  000224R       002 P.LNK   000000          S.FRK   000016              T100Q   000660RG    002 ZF.SLI= 010000
NTIBSY  000212R       002 P.MAIN  000012          S.ITM   000007              T2.ABO= 000100                ZF.TIM= 000200
NTIERB  000243R       002 P.NAM   000004          S.LEN   000004              T2.AST= 100000                ZF.X3P= 000000
NTIERN  000230R       002 P.OWN   000026          S.LHD   000000              T2.CAF= 000400                ZS.ASN= 100000
NTIGRQ  000060R           P.PRI   000002          S.NMST  000002              T2.CHK= 020000                ZS.BSY= 140000
NTINIT  000000R           P.REL   000014          S.OWNR  000003              T2.CKD= 010000                Z.AVL   000014
NTIRCV  000124R       002 P.SIZE  000016          S.PKT   000014              T2.DST= 040000                Z.DAT   000016
NTIRCX  000016R       002 P.STAT  000030          S.PRI   000004              T2.FXD= 002000                Z.DSP   000000
NTIRDA  001042R           P.SUB   000010          S.RCNT  177772              T2.HLT= 000200                Z.FLG   000010
NTIRDT  001100R           P.SWSZ  000022          S.ROFF  177773              T2.REX= 001000                Z.LEN = 000016
NTIRDX  001072R           P.TCB   000026          S.STS   000011              T2.SEF= 004000                Z.LLN   000006
NTISEN  000172RG      002 P.WAIT  000020          S.VCT   000005              T2.SPN= 000004                Z.MAP   000020
```

```
Symbol table
```

```
     55                                          .SBTTL  DATA
     56
     57                                  ;
     58                                  ; LIBRARY MACROS
     59                                  ;
     60                                          .MCALL  NHWDF$,CLKDF$
     61
     62 000000                                   NHWDF$                          ;HARDWARE DEFINITIONS
     63 000000                                   CLKDF$                          ;DEFINE CLOCK QUEUE ENTRY OFFSETS
     64
     65                                  ;
     66                                  ; LOCAL DATA
     67                                  ;
     68
     69         000001                   T5      = 1                             ; 50 MSEC CLOCK BLOCK QUEUED
     70         000002                   T1      = 2                             ; 1 SEC CLOCK BLOCK QUEUED
     71
     72 000000  000000                   TIFLG:  .WORD   0                       ; TIMER BLOCK FLAG WORD
     73
     74
     75                                  ;
     76                                  ; START OF PURE CODE
     77                                  ;
     78 000002                                   .PSECT
```

```
005600  011300                    150$:   MOV     (R3),R0            ; GET ALLOCATION ADDRESS
005602  005023                            CLR     (R3)+              ; ZERO IT
005604  011301                            MOV     (R3),R1            ; GET ALLOCATION SIZE
005606  005023                            CLR     (R3)+              ; ZERO IT
005610  032701  000001                    BIT     #1,R1              ; 16 BIT ALLOCATION?
005614  001003                            BNE     160$               ; IF NE, NO
005616                                    CALL    .DEA16             ; DE-ALLOCATE SUB-ALLOCATION
005622  000404                            BR      170$
005624  000241                    160$:   CLC                        ; DISCARD 16/18 BIT FLAG
005626  006001                            ROR     R1                 ;       ...
005630                                    CALL    $DEA18             ; DE-ALLOCATE SUB-ALLOCATION
005634  005367  000116'           170$:   DEC     COUNT              ; DO IT <ALLOCATION COUNT> TIMES
005640  001357                            BNE     150$               ;       ...
005642  012667  000116'           180$:   MOV     (SP)+,COUNT        ; GET NUMBER OF LIBRARIES
005646  001411                            BEQ     200$               ; IF ZERO, NO WORK TO DO
005650  011300                    190$:   MOV     (R3),R0            ; GET LIBRARY BLOCK BIAS
005652  005023                            CLR     (R3)+              ; ZERO IT
005654  011302                            MOV     (R3),R2            ; GET LIBRARY BLOCK ADDRESS
005656  005023                            CLR     (R3)+              ; ZERO IT

005660                                    CALL    $XLBR              ; RELEASE LIBRARY REFERENCE
005664  005367  000116'                   DEC     COUNT              ; DO IT <LIBRARY COUNT> TIMES
005670  001367                            BNE     190$               ;       ...

005672  022767  000014' 000112'   200$:   CMP     #BLOCK+2,UPTR      ; WAS OUR LINE TABLE THE FIRST ENTRY?
005700  001061                            BNE     250$               ; IF NE, NO
005702  022703  000112'                   CMP     #BLOCK+100,R3      ; DID IT FILL THE ENTIRE 32. WORD BLOCK?
005706  001402                            BEQ     210$               ; IF EQ, YES
005710  005713                            TST     (R3)               ; ANYTHING SIGNIFICANT IN REST OF BLOCK?
005712  001054                            BNE     250$               ; IF NE, YES
005714  016700  000006'           210$:   MOV     UBIAS,R0           ; GET UNLOAD BLOCK ADDRESS
005720  016701  000000G                   MOV     $UNBSZ,R1          ;  AND SIZE
005724                                    CALL    $DEA18             ; DE-ALLOCATE UNLOAD BLOCK
005730  026767  000000C 000006'           CMP     .CXUNL+$NTLHB,UBIAS ; WAS OUR BLOCK FIRST IN LIST ?
005736  001004                            BNE     220$               ; NO
005740  016767  000012' 000000C           MOV     BLOCK,.CXUNL+$NTLHB ; POINT AT NEXT BLOCK
005746  000472                            BR      300$               ; DONE

005750  012767  000100  000000G   220$:   MOV     #100,$RSIZE        ; ASSUME UNMAPPED ALLOCATION SIZE
005756  012703  000000G                   MOV     #$RADDR,R3         ; SET UNMAPPED I/O POINTER
005762  016713  000000C                   MOV     .CXUNL+$NTLHB,(R3) ; ...
005766  005067  000000G                   CLR     $RBLCK             ; ...

005772  012703  000000G                   MOV     #$RBLCK,R3         ; POINT
005776  005067  000000G                   CLR     $RADDR             ; NO OFFSET WITHIN BLOCK
006002  016713  000000C                   MOV     .CXUNL+$NTLHB,(R3) ; ...

006006                            230$:   GETAD                      ; READ IN BLOCK IMAGE
006012  026767  000000G 000006'           CMP     $BFR,UBIAS         ; DOES THIS BLOCK POINT AT US ?
006020  001403                            BEQ     240$               ; YES

006022  016713  000000G                   MOV     $BFR,(R3)          ; SET NEW BIAS/ADDRESS
006026  000767                            BR      230$               ;  AND LOOP

006030  016767  000012' 000000G   240$:   MOV     BLOCK,$BFR         ; POINT AT NEXT BLOCK
006036                                    PUTAD                      ; REWRITE IMAGE
006042  000434                            BR      300$
```

```
                                                                    G  2
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31     PAGE 7
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE              REFERENCES

.TLGTH  = ******  GX        21-1264    21-1271    25-1849
.TSKHD  = ******  GX        25-1849
..PDV     000202 R         #6-188     *8-370     *12-573    *14-744     14-753    *14-766     14-774     18-1149    24-1810
```

```
                                                                    H  2
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31     PAGE 8
MACRO CROSS REFERENCE                                    CREF    04.00
```

```
    306 000770                                          RETURN                                                                                             ;[TD001]
```

G 4

```
**FILE**ID**VVCT




VV      VV   VV      VV      CCCCCCCC   TTTTTTTTTT
VV      VV   VV      VV      CCCCCCCC   TTTTTTTTTT
VV      VV   VV      VV    CC               TT
VV      VV   VV      VV    CC               TT
VV      VV   VV      VV    CC               TT
VV      VV   VV      VV    CC               TT
VV      VV   VV      VV    CC               TT
VV      VV   VV      VV    CC               TT
VV      VV   VV      VV    CC               TT
VV      VV   VV      VV    CC               TT
  VV  VV       VV  VV      CC               TT            ....
  VV  VV       VV  VV      CC               TT            ....
    VV           VV          CCCCCCCC       TT            ....
    VV           VV          CCCCCCCC       TT            ....



LL              SSSSSSSS   TTTTTTTTTT
LL              SSSSSSSS   TTTTTTTTTT
LL            SS               TT
LL            SS               TT
LL            SS               TT
LL            SS               TT
LL              SSSSSS         TT
LL              SSSSSS         TT
LL                    SS       TT
LL                    SS       TT
LL                    SS       TT
LL                    SS       TT
LLLLLLLLLL    SSSSSSSS         TT
LLLLLLLLLL    SSSSSSSS         TT
```

H 4

VVCT - VNP VECTOR SETUP ROUTINE MACRO V05.03b Saturday 29-Jun-85 03:12 Page 4

```
**FILE**ID**VXIT









VV      VV   XX      XX     IIIIII     TTTTTTTTTT
VV      VV   XX      XX     IIIIII     TTTTTTTTTT
VV      VV   XX      XX       II          TT
VV      VV   XX      XX       II          TT
VV      VV     XX  XX         II          TT
VV      VV     XX  XX         II          TT
VV      VV       XX           II          TT
VV      VV       XX           II          TT
VV      VV     XX  XX         II          TT
VV      VV     XX  XX         II          TT
  VV  VV     XX      XX       II          TT          ....
  VV  VV     XX      XX       II          TT          ....
    VV       XX      XX     IIIIII        TT          ....
    VV       XX      XX     IIIIII        TT          ....






LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```

                        .SBTTL  DELIBV - DEALLOCATE LIBRARY VECTOR BLOCK
                ;+
                ; DELIBV - DEALLOCATE LIBRARY VECTOR BLOCK
                ;
                ; INPUTS:
                ;       R0 - LIBRARY BIAS
                ;
                ; OUTPUTS:
                ;       LIBRARY VECTOR BLOCK DEALLOCATED IF PRESENT
                ;-
                        .IF DF  R$$MPL                                                  ;[MP01]

                DELIBV: CALL    $SAVAL          ; SAVE REG'S
                        MOV     R0,R4           ; COPY BIAS
                        MOV     #$NTLHB+.CXVEC,R1 ; GET VECTOR BLOCK ADDRESS
                        MOV     (R1),R0         ; GET FIRST BLOCK
                        BEQ     20$             ; IF NONE - BRANCH

                5$:     CEACC$  R0,R2           ; MAP TO BLOCK
                        GETAD   R2,#V.LEN       ; READ BLOCK

                        BIT     #VF.LIB,$BFR+V.FLG ; LIBRARY VECTOR BLOCK ?
                        BEQ     7$              ; IF NO - BRANCH

                        CMP     $BFR+V.BIAS,R4   ; IS IT OUR VECTOR BLOCK ?
                        BEQ     10$             ; IF YES - BRANCH

                7$:     MOV     R0,R1           ; CURRENT BLOCK BECOMES PREVIOUS BLOCK
                        MOV     $BFR,R0         ; GET NEXT BLOCK
                        BNE     5$              ; CONTINUE

                        RETURN

                10$:    CMP     #$NTLHB+.CXVEC,R1 ; IS PREVIOUS BLOCK THE LISTHEAD ?
                        BNE     15$             ; IF NOT - BRANCH
                        MOV     $BFR,(R1)       ; IF YES - UPDATE LISTHEAD
                        BR      18$             ; CONTINUE

                15$:    MOV     R1,-(SP)        ; COPY UNMAPPED ADDRESS OF PREVIOUS BLOCK
                        MOV     R0,-(SP)        ; COPY UNMAPPED ADDRESS OF CURRENT BLOCK
                        CALL    $XLINK          ; UNLINK VECTOR BLOCK

                18$:    MOV     #V.LEN,R1       ; DEALLOCATE VECTOR BLOCK
                        CALL    $DEPOL          ; ...

                20$:    RETURN

                        .ENDC                                                           ;[MP01]

        000001          .END
```

G 7

```
NTICEX  MACRO V05.03b Saturday 29-Jun-85 03:25
Table of contents
```

```
     5-   53    DATA
     6-  106    SET CEX FUNCTION
     7-  160    LOCAL INITIALIZATION
```

H 7

```
NTICEX    CREATED BY  MACRO  ON 29-JUN-85 AT 03:25     PAGE 4

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME      REFERENCES

CALL             6-140      6-148      6-149      6-151      6-155      7-188      7-202      7-204      7-207      7-227
                 7-250      7-270      7-279
CCBDF$          #5-65       5-72
CLEF$S          #5-64
DHBDF$          #5-65       5-76
DIR$            #5-64
EMSG$R          #5-64       6-156
NHWDF$          #5-65       5-74
NTLER$          #5-64       5-90
PDVDF$          #5-65       5-75
RETURN           6-152      7-210      7-235      7-281
RQST$           #5-64
SDAT$           #5-64
SLTDF$          #5-65       5-73
SWSTK$           7-202      7-227
WTSE$S          #5-64
```

```
**FILE**ID**NTICSR
```

Table of contents

```
     5-   54     MACROS
     6-   81     LOCAL DATA
     7-  107     ERROR MESSAGES
     8-  124     SET EXECUTOR ON
     9-  300     FNDPLI - GET PLI'S PDV INDEX
```

```
                                                                                        G 10
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:26  Page 9-4
Symbol table

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000650    001   (RW,I,LCL,REL,CON)
DATA    000372    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 123
Work  file writes: 126
Size of work file: 29428 Words  ( 115 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:33.40
SY:NTILON.V2,[132,134]NTILON/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,10]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTILON
```

```
                                                                                        H 10
NTILON     CREATED BY  MACRO  ON 29-JUN-85 AT 03:26        PAGE 1
```

```
                                                .SBTTL  ENABLE A LINE
                                        ;+
                                        ; **-ENALIN-ENABLE A LINE IF OWNED BY SPECIFIED LLC
                                        ;
                                        ; IF SO, ENABLE THE LINE BY QUEUEING A CCB TO THE PROPER LLC
                                        ;
                                        ; INPUTS:
                                        ;       R0 - PDV ADDRESS OF ASSOCIATED DDM
                                        ;       R2 - POINTER TO LINE'S ENTRY IN THE REVERSE MAPPING TABLE
                                        ;
                                        ; OUTPUTS:
                                        ;       START FUNCTION QUEUED TO OWNING LLC
                                        ;
                                        ;-

000610                                  ENALIN: SAVRG   <R0,R5>                 ;SAVE R3,R5
000614  010005                                  MOV     R0,R5                   ;GET PDV ADDRESS OF DDM
000616  005001                                  CLR     R1                      ;GET OWNING LLC PDV INDEX
000620  156201  000001                          BISB    1(R2),R1                ; ...
000624  111203                                  MOVB    (R2),R3                 ;GET CHANNEL #
000626  001455                                  BEQ     35$                     ;IF ZERO, BAD CHANNEL
000630  017700  000000G                         MOV     @PDVTA,R0               ;POINT TO PDV INDEX TABLE
000634  060100                                  ADD     R1,R0                   ;ADD OFFSET FOR SPECIFIED LLC
000636  011000                                  MOV     (R0),R0                 ;POINT TO LLC'S PDV

                                                .IF NDF R$$11S
000640  032765  002000  000010                  BIT     #ZF.PSE,Z.FLG(R5)       ; IS THIS A DLM CIRCUIT?
000646  001402                                  BEQ     10$                     ; BR IF NO
000650                                          CALL    DLMSET                  ; SET UP DATA LINK MAPPING PARAMETERS
                                                .ENDC

000654  016000  000016                  10$:    MOV     Z.DAT(R0),R0            ;GET POINTER TO LLC'S DATABASE
000660  001440                                  BEQ     35$                     ;IF ZERO, LLC NOT LOADED!!!

000662                                  15$:    SWSTK$  30$                     ;; ENTER SYSTEM STATE
000666                                          CALL    @CCBGT                  ;; TRY TO ALLOCATE A CCB
000672  103420                                  BCS     20$                     ;; IF CS, ALLOCATION FAILURE
000674  005064  000014                          CLR     C.BUF1(R4)              ;; CLEAR FLAGS WORD
000700  110164  000007                          MOVB    R1,C.STA(R4)            ;; SET UP PDV AND
000704  110364  000006                          MOVB    R3,C.LIN(R4)            ;;   CHANNEL # TO TURN ON
000710  005064  000004                          CLR     C.NSP(R4)               ;; CONTROL IS RETURNED TO AUX
000714  012764  000004  000012                  MOV     #STRT,C.STS(R4)         ;; TURN ON THE LINE
000722  012764  001006  000010                  MOV     #FC.CTL!FS.STR,C.FNC(R4);; SET UP FUNCTION CODE
000730                                          CALLR   @LLCRS                  ;; QUEUE CCB TO LLC
                                                                                                        ;**-1
000734                                  20$:    RETC    R0                      ;; SET USER C-BIT
000746                                          RETURN                          ;; LEAVE SYSTEM STATE

000750  103005                          30$:    BCC     40$                     ; IF CC, REQUEST QUEUED
000752                                          WSIG$S                          ; ALLOCATION FAILURE ... WAIT A BIT
000760  000740                                  BR      15$                     ;   THEN TRY AGAIN

000762  000261                          35$:    SEC                             ; INDICATE LINE NOT ENABLED
000764                                  40$:    RESRG   <R5,R0>                 ; RESTORE R5,R3
000770                                          RETURN
```

```
                                .TITLE  NTIMFL
                                .IDENT  /V05.00/

                        ;
                        ; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                        ; TRANSFERRED.
                        ;
                        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                        ; CORPORATION.
                        ;
                        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                        ;
                        ; MODULE DESCRIPTION:
                        ;
                        ;       NTINIT - MARKED-FOR-LOAD SCAN
                        ;
                        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                        ;
                        ; IDENT HISTORY:
                        ;
                        ; 1.00  27-FEB-78
                        ;       VERSION 2.0 RELEASE
                        ;
                        ; 2.00  14-DEC-79
                        ;       DECNET-11M/S V3.0
                        ;       DECNET-11M-PLUS V1.0
                        ;
                        ; 3.00  16-APR-82
                        ;       DECNET-11M V3.1
                        ;       DECNET-11M-PLUS V1.1
                        ;
                        ; 4.00  07-NOV-83
                        ;       DECNET-11M V4.0
                        ;       DECNET-11M-PLUS V2.0
                        ;
                        ; 5.00  22-JUL-85
                        ;       DECnet-11M/S V4.2
                        ;       DECnet-11M-Plus V3.0
                        ;       DECnet-Micro/RSX V1.0
                        ;
                        ;

```

```
**FILE**ID**NTINIT




NN      NN    TTTTTTTTTT    IIIIII    NN      NN    IIIIII    TTTTTTTTTT
NN      NN    TTTTTTTTTT    IIIIII    NN      NN    IIIIII    TTTTTTTTTT
NN      NN        TT          II      NN      NN      II          TT
NN      NN        TT          II      NN      NN      II          TT
NNNN    NN        TT          II      NNNN    NN      II          TT
NNNN    NN        TT          II      NNNN    NN      II          TT
NN  NN  NN        TT          II      NN  NN  NN      II          TT
NN  NN  NN        TT          II      NN  NN  NN      II          TT
NN    NNNN        TT          II      NN    NNNN      II          TT
NN    NNNN        TT          II      NN    NNNN      II          TT
NN      NN        TT          II      NN      NN      II          TT        ....
NN      NN        TT          II      NN      NN      II          TT        ....
NN      NN        TT        IIIIII    NN      NN    IIIIII        TT        ....
NN      NN        TT        IIIIII    NN      NN    IIIIII        TT        ....




LL                SSSSSSSS    TTTTTTTTTT
LL                SSSSSSSS    TTTTTTTTTT
LL            SS                  TT
LL            SS                  TT
LL            SS                  TT
LL            SS                  TT
LL                SSSSSS          TT
LL                SSSSSS          TT
LL                      SS        TT
LL                      SS        TT
LL                      SS        TT
LL                      SS        TT
LLLLLLLLLL    SSSSSSSS            TT
LLLLLLLLLL    SSSSSSSS            TT
```

```
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:02
Table of contents
```

```

                .SBTTL  VECSET - SET UP PROCESS VECTOR TABLE
        ;+
        ; VECSET - SET UP PROCESS VECTOR TABLE
        ;
        ; INPUTS:
        ;       .CXVEC - LIST OF VECTOR BLOCKS
        ;
        ; OUTPUTS:
        ;       PROCESS VECTOR TABLES TRANSLATED
        ;
        ;-
                .IF DF  R$$MPL

        VECSET: SWSTK$  20$                     ; ENTER SYSTEM STATE
                SAVMAP                          ; SAVE MAPPING
                MOV     @NTLPT,@KSAR6           ; MAP TO NTL HOME BLOCK
                MOV     140000+.CXVEC,R5        ; GET VECTOR BLOCK ADDRESS
                BEQ     15$                     ; IF NONE- BRANCH
                MOV     R5,VECLHD               ; SAVE LISTHEAD FOR DEALLOCATION
                CLR     140000+.CXVEC           ; INIT VECTOR BLOCK LISTHEAD

        5$:     MOV     R5,-(SP)                ; SAVE UNMAPPED POOL.. ADDRESS
                CALL    @CEACC                  ; ...
                MOV     (SP)+,R5                ; GET MAPPED VIRTUAL ADDRESS
                MOV     V.ADD(R5),R3            ; COPY TABLE ADDRESS OF PROCESS
                BIC     #160000,R3              ; BIAS FOR APR6
                BIS     #140000,R3              ; ...
                MOV     V.SIZ(R5),R2            ; COPY TABLE LENGTH
                MOV     V.PDV(R5),R4            ; GET PDV ADDRESS OR LIBRARY TABLE BIAS
                SAVMAP                          ; SAVE MAPPING TO VECTOR BLOCKS

                BIT     #VF.LIB,V.FLG(R5)       ; LIBRARY VECTORED BLOCK ?
                BEQ     6$                      ; IF NO - BRANCH
                MOV     R4,@KSAR6               ; MAP TO LIBRARY
                BR      9$                      ; CONTINUE

        6$:     MOV     Z.DSP(R4),@KSAR6        ; MAP TO PROCESS

        9$:     MOV     @#112,R1                ; COPY BIAS OF EXEC ROUTINE WHICH
                                                ; CALLS THE VECTOR TRANSLATE ROUTINE
                MOV     (R1),R1                 ; GETTING TO ACTUAL BIAS
                MOV     (R1),-(SP)              ; PUT BIAS ON STACK FOR $MPPRO
                MOV     #120000,-(SP)           ; MAP THRU APR5
                CALL    @MPPRO                  ; CALL DIRECTIVE COMMON ROUTINE THE LONG WAY
                RESMAP                          ; RESTORE MAPPING TO VECTOR BLOCKS

                MOV     (R5),R5                 ; GET NEXT BLOCK
                BNE     5$                      ; IF MORE - BRANCH

                MOV     VECLHD,R5               ; DEALLOCATE BLOCKS (AUX VECTOR TABLE IS
                                                ; NOW SET UP)
        10$:    MOV     R5,-(SP)                ; COPY UNMAPPED ADDRESS
                MOV     (SP),-(SP)              ; SAVE UNMAPPED ADDRESS
                CALL    @CEACC                  ; MAP TO BLOCK
                MOV     (SP)+,R5                ; COPY VIRTUAL ADDRESS OF BLOCK
```

```
                                                                                       G 15
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:02  Page 17-3
Symbol table

Z.NAM   000004               $ERR04  000462R      002 $NIFLG  000226RG     002 $PUMR = ****** GX        .BASEB= 140000
Z.PCB   000012               $ERR12  000520RG     002 $NIFNC  000204RG     002 $QASTT= ****** GX        .CXALL  000062 G
Z.SCH   000007               $FMASK= ****** GX        $NILON= ****** GX        $SLTMA= ****** GX        .CXCSR  000004 G
$ABTIM= ****** GX            $HEADR= ****** GX        $NIMRK  000070R      002 $SLTNM= ****** GX        .CXLBR  000060 G
$ALOCB= ****** GX            $LDPWF= ****** GX        $NINTL  000376RG         $SRSTD= ****** GX        .CXLB1  000030 G
$CALLX= ****** GX            $LGSTT= ****** GX        $NIPRM  000122R      002 $TKPS = ****** GX        .CXLB2  000044 G
$CBDMG= ****** GX            $LLCAL= ****** GX        $NIRDB  000032RG     002 $TKTCB= ****** GX        .CXLEN  000064 G
$CCBGT= ****** GX            $LLCRS= ****** GX        $NIRNL  000134RG     002 $TK100= ****** GX        .CXPCB  000006 G
$CEACC= ****** GX            $LLCTA= ****** GX        $NIRTN  000026RG     002 $T1SCL= ****** GX        .CXSYM  000010 G
$CLKHD= ****** GX            $NERDD  001014RG         $NISNL  000152RG     002 $T100C= ****** GX        .CXUNL  000024 G
$C5TA = ****** GX            $NERDN  000772RG         $NIXPT  000002RG     002 $T100Q= ****** GX        .CXVEC  000026 G
$DBECL  000004RG     002     $NER50  000750RG         $NIX25= ****** GX        $UMRPT= ****** GX        .CXVSM  000016 G
$DBXPT  000006RG     002     $NIAST  000120R      002 $NTLFC  000164RG     002 $$$   = 000062           .NTGCD  000000
$DEACB= ****** GX            $NICEX= ****** GX        $NTLPT= ****** GX        $$$ARG= 000002           .NTPCB  000000 G
$DECPT= ****** GX            $NIECL  000000RG     002 $PDVID= ****** GX        $$$OST= 000012           .NTUMR  000002 G
$DEVHD= ****** GX            $NIERA  000010RG     002 $PDVNM= ****** GX        $$$T1 = 000006           .SBPCB  000066 G
$ERR02  000352R      002     $NIERR  000576RG         $PDVTA= ****** GX        $$$$  = 000000           .$$$$.= 000034
$ERR03  000430R      002     $NIEVC  000102RG     002 $PSIPT= ****** GX

. ABS.  177776    000    (RW,I,GBL,ABS,OVR)
        001334    001    (RW,I,LCL,REL,CON)
DATA    000714    002    (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 117
Work  file writes: 109
Size of work file: 25359 Words  ( 100 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:02:06.12
DB2:NTINIT.T34,[132,134]NTINIT/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTINIT
```

```
                                                                                       H 15
NTINIT     CREATED BY  MACRO  ON 5-SEP-85 AT 11:04        PAGE 1
```

```
                                                   .SBTTL  START CEX TIMERS

                                          ;+
                                          ; $NITIM - START CEX TIMERS IF NOT ALREADY STARTED
                                          ;
                                          ; INPUTS:
                                          ;        NONE
                                          ;
                                          ; OUTPUTS:
                                          ;        IF THE TIMERS ARE ALREADY IN THE CLOCK QUEUE, THEN THIS IS A
                                          ;          NO-OP, ELSE THE TIMERS ARE STARTED.
                                          ;        R0 IS DESTROYED.
                                          ;-

 000002  016700  000000G                  $NITIM::MOV     CLKHD,R0                ;POINT AT THE CLOCK QUEUE

 000006                                           SWSTK$  30$                     ;* ENTER SYSTEM STATE

 000012                                           MTPS    #PR7                    ;;;* DISABLE INTERUPTS
 000020  011000                           10$:    MOV     (R0),R0                 ;;;* POINT AT NEXT BLOCK
 000022  001416                                   BEQ     20$                     ;;;* IF EQ, END
 000024  026700  000000G                          CMP     T100C,R0                ;;;* ENTRY FOR 50 MSEC TIMER?
 000030  001004                                   BNE     12$                     ;;;* IF NO - BRANCH
 000032  052767  000001  177740                   BIS     #T5,TIFLG               ;;;* IF YES - SET 50 MSEC FLAG
 000040  000767                                   BR      10$                     ;;;* CONTINUE

 000042  026700  000000G                  12$:    CMP     T1SCL,R0                ;;;* ENTRY FOR 1 SEC TIMER?
 000046  001364                                   BNE     10$                     ;;;* IF NE, NO - CHECK NEXT BLOC
 000050  052767  000002  177722                   BIS     #T1,TIFLG               ;;;* IF YES - SET 1 SEC FLAG
 000056  000760                                   BR      10$                     ;;;* CONTINUE

 000060                                   20$:    MTPS    #0                      ;;;* RE-ENABLE INTERRUPTS
 000064  032767  000001  177706                   BIT     #T5,TIFLG               ;* START 50 MSEC TIMER ?
 000072  001002                                   BNE     25$                     ;* IF NO - BRANCH
 000074                                           CALL    T50IN                   ;* START 50 MSEC TIMER (LOCAL CODE)
 000100  032767  000002  177672           25$:    BIT     #T1,TIFLG               ;* START 1 SEC TIMER ?
 000106  001002                                   BNE     30$                     ;* IF NO - BRANCH
 000110                                           CALL    T1SIN                   ;* START 1 SEC TIMER (LOCAL CODE)

 000114                                   30$:    RETURN                          ;* RETURN TO USER AND THEN TO CALLER
```

```

006044  012700  000112'          250$:   MOV     #BLOCK+100,R0   ; GET END OF BLOCK ADDRESS
006050  020003                           CMP     R0,R3           ; ARE WE THERE NOW?
006052  001413                           BEQ     280$            ; IF EQ, YES
006054  010001                           MOV     R0,R1           ; CALCULATE SIZE REMAINING IN BLOCK
006056  160301                           SUB     R3,R1           ;       ...
006060  006201                           ASR     R1              ;       ...
006062  016702  000112'                  MOV     UPTR,R2         ; COPY REST OF BLOCK ONTO WHAT USED TO BE
006066  012322                   260$:   MOV     (R3)+,(R2)+     ; OUR LINE TABLE ENTRY
006070  005301                           DEC     R1
006072  001375                           BNE     260$            ; ...
006074  005022                   270$:   CLR     (R2)+           ; ZERO THE REST OF THE BLOCK
006076  020002                           CMP     R0,R2           ;       ...
006100  001375                           BNE     270$            ;       ...
006102                           280$:
006102                                   SWTCHO  #BLOCK          ; USE BLOCK BUFFER
006110  016767  000006' 000000G          MOV     UBIAS,$RBLCK    ; SET BIAS FOR MAPPED SYSTEM
006116  005067  000000G                  CLR     $RADDR          ; CLEAR OFFSET
006122                                   PUTAD   ,#100           ; REWRITE UNLOAD BLOCK

006134                           300$:   SWTCHO                  ; USE DEFAULT BUFFERS
006142                                   RETURN


```

```
                                                                        H  2
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31       PAGE 8
MACRO CROSS REFERENCE                                      CREF    04.00
MACRO NAME      REFERENCES

ADJDF$          #5-62       5-77
ANBDF$          #5-60       5-81
ASR$            #5-59       10-449     10-463     10-476     10-493     11-531     17-1031
CALL             7-207      7-208      7-231      7-234      8-322      8-334      8-349      8-353      8-374      8-379
                 8-382      8-383      10-441     10-442     10-452     10-453     10-465     10-466     10-472     10-478
                 10-479     10-485     10-489     10-496     10-497     10-502     11-524     11-533     12-563     12-576
                 12-584     12-589     13-618     13-626     14-652     14-654     14-686     14-700     14-737     14-739
                 14-743     14-747     14-749     14-751     14-752     14-759     14-765     14-769     14-777     14-808
                 15-827     15-836     17-962     17-1007    17-1020    17-1043    17-1050    17-1065    18-1117    20-1204
                 20-1213    20-1226    20-1234    21-1262    21-1264    21-1271    21-1281    21-1284    21-1294    21-1308
                 21-1309    21-1323    21-1332    21-1340    21-1345    21-1346    21-1348    21-1352    21-1355    21-1361
                 21-1367    21-1373    21-1380    21-1390    21-1394    21-1429    21-1431    21-1469    21-1484    21-1494
                 21-1496    21-1499    21-1509    21-1515    21-1517    21-1528    21-1538    21-1545    21-1547    21-1552
                 21-1565    21-1567    21-1571    22-1601    23-1654    23-1664    23-1686    23-1707    23-1711    23-1721
                 23-1733    23-1748    23-1756    23-1776    24-1814    24-1819    24-1826    25-1849    25-1856
CALLR           #5-60       8-266      14-657     21-1369    21-1375
CEACC$          #5-59       7-207      21-1494    21-1515    21-1545    21-1565
CRASH            8-393      8-394      12-597     12-598     14-692     14-786     14-787     15-852     17-1063    17-1064
                 21-1460    23-1657
CTRDF$          #5-61       5-85
DHBDF$          #5-61       5-87
ECDDB$          #5-61       5-83
ERBLK$          #5-60      #8-262     #12-558     13-628     17-1068    20-1236   #21-1255
ERMSG$          #5-60      #8-262     #12-558     13-627     17-1066    17-1067    20-1235   #21-1255
ERRPT$          #5-60      #8-262     #12-558     13-626     17-1065    20-1234   #21-1255
FLTDF$          #5-60       5-80
GETAD           #5-59       7-208      18-1117    21-1496    21-1517    21-1547    21-1567    22-1601    23-1748
GETRC           #5-61
GETRV           #5-59       8-322      8-334      10-442     10-453     10-466     10-479     10-485     10-497     11-524
                 14-700     15-827     15-836     17-962     17-1007    21-1264    21-1323    21-1469    21-1509    21-1538
                 24-1814    25-1849    25-1856
NKRDF$          #5-70       5-71
OBJDF$          #5-60       5-79
PCBDF$          #5-61       5-88
PLBDF$          #5-61       5-84
PUTAD           #5-59       20-1226    23-1756    23-1776
PUTRC           #5-59       17-1050    21-1271    21-1394    21-1484
RAND            #7-208      7-208     #8-322      8-322     #8-334      8-334     #10-442     10-442    #10-453     10-453
                #10-466     10-466    #10-479     10-479    #10-485     10-485    #10-497     10-497    #11-524     11-524
                #14-700     14-700    #15-827     15-827    #15-836     15-836    #17-962     17-962    #17-1007    17-1007
                #17-1050    17-1050   #18-1117    18-1117   #20-1226    20-1226   #21-1264    21-1264   #21-1271    21-1271
                #21-1323    21-1323   #21-1394    21-1394   #21-1469    21-1469   #21-1484    21-1484   #21-1496    21-1496
                #21-1509    21-1509   #21-1517    21-1517   #21-1538    21-1538   #21-1547    21-1547   #21-1567    21-1567
                #22-1601    22-1601   #23-1748    23-1748   #23-1756    23-1756   #23-1776    23-1776   #24-1814    24-1814
                #25-1849    25-1849   #25-1856    25-1856
RESRG           #5-62
RETURN           7-222      7-236      8-388      9-424      10-504     11-534     12-592     13-621     14-781     14-795
                 14-809     15-850     15-879     17-1058    18-1166    19-1188    20-1228    21-1259    21-1455    21-1485
                 21-1503    21-1532    21-1556    21-1576    22-1642    23-1779    24-1829    25-1861
RNBDF$          #5-60       5-82
SAVRG           #5-62
SCBDF$          #5..59      5-76
```

```
                                                                        I  2
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31       PAGE 9
```

```
                                                .SBTTL  XLDTE - DEALLOCATE LOCAL DTE BLOCK

                                        ;+
                                        ;
                                        ; XLDTE - DEALLOCATE LOCAL DTE DESCRIPTOR BLOCK
                                        ;
                                        ; INPUTS:
                                        ;       NONE
                                        ;
                                        ; OUTPUTS:
                                        ;       LOCAL DTE DESCRIPTOR BLOCKS, CHANNEL HASH TABLES, AND
                                        ;       OUTGOING CHANNEL LISTS ARE DEALLOCATED
                                        ;
                                        ;-

000772                                  XLDTE:
000772                                          CALL    $SAVAL          ; Save all registers                            ;[TD001]
000776                                          SAVMAP                  ; Save previous mapping                         ;[TD001]
                                                                                                                        ;**-2
001002                                          SWTCHI  #LSTHBF         ; USE LOCAL BUFFER FOR LISTHEAD
001010  016705  000000G                         MOV     $PSIPT,R5       ; GET PSI HOME BLOCK ADDRESS
001014  062705  000002                          ADD     #H$LDTE,R5      ; POINT TO LISTHEAD
001020                                          GETRV   R5,#2           ; READ IN LISTHEAD
001036                                          SWTCHI                  ; USE DEFAULT BUFFER
001044  016700  000000'                 10$:    MOV     LSTHBF,R0       ; GET NEXT BLOCK IN LIST
001050  001467                                  BEQ     40$             ; BR IF END OF LIST
001052                                          CEACC$  R0,R4           ; CONVERT TO MAPPED ADDRESS
001062                                          GETAD   R4,#L$LEN       ; Read fixed part of descriptor                 ;[TD001]
001100  016767  000000G 000000'                 MOV     $BFR,LSTHBF     ; Save link address                             ;[TD001]
                                                                                                                        ;[TD001]
001106                                          SAVRG   <R0>            ; Save R0 (unmapped address of block)           ;[TD001]
001110  012700  000122                          MOV     #L$LEN,R0       ; Get length of basic descriptor                ;[TD001]
001114  032767  000000G 000000G                 BIT     #PF.PSV,$PFLAG  ; Check if descriptor extension present         ;[TD001]
001122  001402                                  BEQ     12$             ; Branch if not present                         ;[TD001]
001124  012700  000134                          MOV     #L$GLEN,R0      ; Get length of extended descriptor             ;[TD001]
001130                                  12$:    SAVRG   <R0>            ; Stack length to deallocate                    ;[TD001]
                                                                                                                        ;**-5
001132  016700  000034G                 15$:    MOV     L$CHLS+$BFR,R0  ; GET OUTGOING CHANNELS LIST ADDRESS
001136  001417                                  BEQ     20$             ; BR IF NONE
001140                                          GETRV   R0,#2           ; Read in listhead (rest of $BFR unchanged)     ;[TD001]
001156  016767  000000G 000034G                 MOV     $BFR,L$CHLS+$BFR; SAVE LINK TO NEXT                             ;**-1
001164  012701  000006                          MOV     #K$LEN,R1       ; GET LENGTH OF OUTGOING CHANNELS LIST
001170                                          CALL    $DEA16          ; DEALLOCATE THE BLOCK
001174  000756                                  BR      15$             ; GO AGAIN

001176  016700  000030G                 20$:    MOV     L$CHTB+$BFR,R0  ; GET HASH TABLE ADDRESS
001202  001405                                  BEQ     30$             ; BR IF NONE
001204  016701  000032G                         MOV     L$CTEN+$BFR,R1  ; GET NUMBER OF ENTRIES IN HASH TABLE
001210                                          ASL$    2,R1            ; CALCULATE SIZE (4 BYTES PER ENTRY)

001214  060116                          25$:    ADD     R1,(SP)         ; INCLUDE LENGTH OF HASH TABLE IN DEALLOCATION
001216                                  30$:    RESRG   <R1>            ; GET LENGTH TO DEALLOCATE
001220                                          RESRG   <R0>            ; RETRIEVE ADDRESS OF LOCAL DTE DESCRIPTOR
001222                                          CALL    $DEPOL          ; DEALLOCATE BLOCK
001226  000706                                  BR      10$             ; DO FOR NEXT BLOCK

001230  005067  000000G                 40$:    CLR     $BFR            ; CLEAN UP LISTHEAD
```

```
                        .TITLE  VVCT - VNP VECTOR SETUP ROUTINES
                        .IDENT  /V05.00/

                ;
                ; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                ;
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                ; TRANSFERRED.
                ;
                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                ; CORPORATION.
                ;
                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                ;
                ; MODULE DESCRIPTION:
                ;
                ;       VNP - INTERRUPT VECTOR SETUP ROUTINES
                ;
                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                ;
                ; IDENT HISTORY:
                ;
                ; 1.00  27-FEB-78
                ;       VERSION 2.0 RELEASE
                ;
                ; 2.00  14-DEC-79
                ;       DECNET-11M/S V3.0
                ;       DECNET-11M-PLUS V1.0
                ;
                ; 3.00  16-APR-82
                ;       DECNET-11M V3.1
                ;       DECNET-11M-PLUS V1.1
                ;
                ; 4.00  07-NOV-83
                ;       DECNET-11M V4.0
                ;       DECNET-11M-PLUS V2.0
                ;
                ; 5.00  22-JUL-85
                ;       DECnet-11M/S V4.2
                ;       DECnet-11M-Plus V3.0
                ;       DECnet-Micro/RSX V1.0
                ;
                ;

```

```
        .TITLE  VXIT - VNP EXIT ROUTINES
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       VNP - TERMINATION ROUTINES
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;


```

```
                                                                                        H 6
VXLB - VNP LIBRARY DEALLOCATION MACRO V05.03b Friday 06-Sep-85 14:39 Page 6-1
Symbol table

A$$CHK= 000000                G$$TSS= 000000                L$$P11= 000001                N$$PEM= 000001                V.BIAS  000004
A$$CPS= 000000                G$$TTK= 000000                L$$11R= 000000                P$$P45= 000000                V.FLG   000002
A$$PRI= 000000                G$$WRD= 000000                M$$CRB= 000124                P$$WRD= 000000                V.LEN   000012
A$$TRP= 000000                I$$RAR= 000000                M$$CRX= 000000                Q$$OPT= 000010                V.LNK   000000
C$$CKP= 000000                I$$RDN= 000000                M$$FCS= 000000                R$$DER= 000000                V.PDV   000004
C$$ORE= 000400                K$$CNT= 177546                M$$MGE= 000000                R$$K11= 000001                V.SIZ   000010
C$$RSH= 177564                K$$CSR= 177546                M$$NET= 000000                R$$SND= 000000                X$$DBT= 000000
D$$BUG= 177514                K$$LDC= 000000                M$$OVR= 000000                R$$11M= 000000                $DEA18= ****** GX
D$$ISK= 000000                K$$TPS= 000074                N$$ACC= 000001                S$$WRG= 000000                $GETLB= ****** GX
D$$L11= 000001                LB.ADR= ****** GX             N$$BUF= 000001                S$$YSZ= 007600                $NTLHB= ****** GX
D$$YNC= 000000                LB.SIZ= ****** GX             N$$LDV= 000001                T$$KMG= 000000                $NTLPT= ****** GX
D$$YNM= 000000                LB.USE= ****** GX             N$$MCP= 000001                T$$MIN= 000000                $PUTLB= ****** GX
E$$XPR= 000000                LD$LP = 000000                N$$MLL= 000001                VF.LIB= 000001                $XLBR   000000RG
F$$LVL= 000001                L$$ASG= 000000                N$$MOV= 000010                V$$CTR= 001000                .CXLBR= ****** GX
G$$TPP= 000000                L$$DRV= 000000                N$$NCT= 000001                V.ADD   000006                .MGMGE= ****** GX


. ABS.  000012    000   (RW,I,GBL,ABS,OVR)
        000170    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 9457 Words  ( 37 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:08.24
DB2:VXLB.T47,[132,134]VXLB/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VXLB
```

```
                                                                I 6
VXLB      CREATED BY  MACRO  ON 6-SEP-85 AT 14:39       PAGE 1
```

```
          .TITLE  NTICEX
          .IDENT  /V05.00/
;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;         NTINIT - ''SET CEX'' PROCESSING
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00    27-FEB-78
;         VERSION 2.0 RELEASE
;
; 2.00    14-DEC-79
;         DECNET-11M/S V3.0
;         DECNET-11M-PLUS V1.0
;
; 3.00    16-APR-82
;         DECNET-11M V3.1
;         DECNET-11M-PLUS V1.1
;
; 4.00    07-NOV-83
;         DECNET-11M V4.0
;         DECNET-11M-PLUS V2.0
;
; 5.00    22-JUL-85
;         DECnet-11M/S V4.2
;         DECnet-11M-Plus V3.0
;         DECnet-Micro/RSX V1.0
;
;
```

```
**FILE**ID**NTICSR









NN      NN  TTTTTTTTTT    IIIIII      CCCCCCCC    SSSSSSSS  RRRRRRRR
NN      NN  TTTTTTTTTT    IIIIII      CCCCCCCC    SSSSSSSS  RRRRRRRR
NN      NN      TT          II      CC          SS          RR      RR
NN      NN      TT          II      CC          SS          RR      RR
NNNN    NN      TT          II      CC          SS          RR      RR
NNNN    NN      TT          II      CC          SS          RR      RR
NN  NN  NN      TT          II      CC            SSSSSS    RRRRRRRR
NN  NN  NN      TT          II      CC            SSSSSS    RRRRRRRR
NN    NNNN      TT          II      CC                  SS  RR  RR
NN    NNNN      TT          II      CC                  SS  RR  RR
NN      NN      TT          II      CC                  SS  RR    RR       ....
NN      NN      TT          II      CC                  SS  RR    RR       ....
NN      NN      TT        IIIIII      CCCCCCCC  SSSSSSSS    RR      RR     ....
NN      NN      TT        IIIIII      CCCCCCCC  SSSSSSSS    RR      RR     ....








LL          SSSSSSSS  TTTTTTTTTT
LL          SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

NTICSR MACRO V05.03b Saturday 29-Jun-85 03:25
Table of contents

```
        .TITLE  NTILON - NTINIT 'EXECUTOR ON' ROUTINE
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       NTINIT - ''SET EXECUTOR STATE ON'' PROCESSING
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                                                            H 10
NTILON      CREATED BY  MACRO  ON 29-JUN-85 AT 03:26     PAGE 1

SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE               REFERENCES

ALOCB    = ******   GX        8-213        8-254
AREAMK   = 176000            #6-86
ASSIGN     000042 R          #6-105        8-150
A.LULU   = 000002             6-105
BUFUMP   = 172354            #5-66
CMODE    = 140000            #5-66
DEACB    = ******   GX        8-288
DECPT    = ******   GX        8-157        8-202        8-216        8-246
D$NBRA     000054             8-247        8-249
D$NLN      000030            *8-158        8-203        8-217
E$NBR      000014            #5-71
E$NBS      000020            #5-71
E$NCR      000034            #5-71
E$NCS      000036            #5-71
E$NIC      000044            #5-71
E$NLEN     000050            #5-71
E$NLLA     000012            #5-71
E$NLNK     000000            #5-71
E$NML      000040            #5-71
E$NMR      000024            #5-71
E$NMS      000030            #5-71
E$NNOD     000002            #5-71
E$NRT      000042            #5-71
E$NRTP     000005            #5-71
E$NSEG     000010            #5-71
E$NTIM     000046            #5-71
E$NUSE     000004            #5-71
E$STRT     000006            #5-71
FNDPDV   = ******   GX        8-141
FNDPLI     000610 R          #9-315
HEADR    = ******   GX        8-164
H.LUN      000076             8-165
I$$AS    = ******             5-66         5-70
KISAR0   = 172340            #5-66
KISAR6   = 172354            #5-66
LONACP     000014 R          #6-104        8-265
LVL1     = 002000            #6-87
MPAR     = 172100            #5-66
MPCSR    = 177746            #5-66
M$$MGE   = 000000             5-66
NF$MOU   = 000040             8-272
NF.END   = 000200             8-175
NF.LV2   = 000100             8-181
N$FLG      000005             8-272
N$PLD      000016             8-200       *8-204        8-207       *8-229        8-236
N$PRI      000076            *8-262
N$$VCT   = ******             9-320
PDVID    = ******   GX        9-317
PDVTA    = ******   GX        8-155
PIRQ     = 177772            #5-66
PLDADR     000000 R          #6-97        *8-233        8-286
PLDLEN     000002 R          #6-98        *8-234        8-287
```

```
                                                            I 10
NTILON      CREATED BY  MACRO  ON 29-JUN-85 AT 03:26     PAGE 2
```

```
                                             .SBTTL  DLMSET - SET UP COUNTER BLOCK FIELDS FOR DLM CIRCUITS

                                    ;+
                                    ;
                                    ; DLMSET - SET UP COUNTER BLOCK FIELDS FOR DLM CIRCUITS
                                    ;
                                    ; INPUTS:
                                    ;        STAADD - ADDRESS OF STATION TABLE
                                    ;        R0 - XPT'S PDV ADDRESS
                                    ;        R2 - ADDRESS OF REVERSE MAPPING TABLE ENTRY
                                    ;
                                    ; OUTPUTS:
                                    ;        DLM FLAGS AND SVC DESCRIPTOR ADDRESS OR PORT NUMBER (PVC) STORED
                                    ;            IN COUNTER BLOCK
                                    ;        COUNTER BLOCK IS ALLOCATED IF IT DOES NOT ALREADY EXIST
                                    ;
                                    ;-
                                             .IF NDF R$$11S
000772                              DLMSET:
000772                                       SWSTK$  60$               ;; ENTER SYSTEM STATE
000776   011204                              MOV     (R2),R4           ;; GET CHANNEL NUMBER AND PDV
001000   017702  000000G                     MOV     @PSIPT,R2         ;; GET POINTER TO PSI HOME BLOCK
001004   001006                              BNE     10$               ;; BR IF HOME BLOCK EXISTS
001006                                       RETC    R2                ;; ELSE SET USER C-BIT
001020   000506                              BR      60$               ;; AND EXIT
001022   010003                     10$:     MOV     R0,R3             ;; POINT TO XPT'S PDV
001024   062703  000020                      ADD     #Z.MAP,R3         ;; POINT TO FORWARD MAPPING
001030   005046                              CLR     -(SP)             ;; GET CHANNEL NUMBER
001032   110416                              MOVB    R4,(SP)           ;; ...
001034   006316                              ASL     (SP)              ;; MAKE IT A WORD INDEX
001036   062603                              ADD     (SP)+,R3          ;; POINT TO SLN AND STATION NUMBER
001040   011304                              MOV     (R3),R4           ;; GET SLN AND STATION NUMBER
001042                                       SAVMAP                    ;; SAVE CURRENT MAPPING
001046   016000  000016                      MOV     Z.DAT(R0),R0      ;; GET ADDRESS OF XPT'S DDB
001052                                       MAP     N$TLC+2(R0)       ;; MAP TO CIRCUIT COUNTERS
001060   016003  000100                      MOV     N$TLC(R0),R3      ;; GET NUMBER OF COUNTERS
001064   016000  000104                      MOV     N$TLC+4(R0),R0    ;; GET ADDRESS OF FIRST COUNTER BLOCK
001070                                       BIAS    R0                ;; BIAS COUNTER BLOCK ADDRESS
001100   005710                     20$:     TST     (R0)              ;; HAS THIS BLOCK BEEN ASSIGNED ALREADY?
001102   100407                              BMI     30$               ;; BR IF NO - USE IT
001104   020410                              CMP     R4,(R0)           ;; ELSE IS THIS THE CORRECT ENTRY?
001106   001405                              BEQ     30$               ;; BR IF YES
001110   062700  000046                      ADD     #T$LLEN,R0        ;; POINT TO NEXT COUNTER BLOCK
001114                                       SOB     R3,20$            ;; CHECK ALL COUNTER BLOCKS
001120   000444                              BR      50$               ;; EXIT
001122   052760  000200  000044     30$:     BIS     #PX$DLM,T$FLAG(R0) ;; SET FLAG TO INDICATE DLM
001130   132777  000002  176642              BITB    #SF.SVC,@STAADD ;; IS THIS AN SVC?
001136   001435                              BEQ     50$               ;; BR IF NO
001140   052760  000100  000044              BIS     #PX$SVC,T$FLAG(R0) ;; ELSE INDICATE SVC
001146                                       SAVMAP                    ;; SAVE CURRENT MAPPING
001152   062702  000036                      ADD     #H$SVC,R2         ;; GET FIRST SVC DESCRIPTOR ADDRESS
001156   011202                     40$:     MOV     (R2),R2           ;; GET NEXT BLOCK IN LIST
001160   001416                              BEQ     45$               ;; BR IF NOT FOUND
001162   010246                              MOV     R2,-(SP)          ;; GET UNMAPPED ADDRESS
001164                                       CALL    @CEACC            ;; CONVERT TO MAPPED ADDRESS
001170   012602                              MOV     (SP)+,R2          ;; RETRIEVE MAPPED ADDRESS
001172   126267  000002  176603              CMPB    S$CTL(R2),CTLNUM;; IS THIS THE CORRECT SVC DESCRIPTOR?
```

```
                                .SBTTL  MACRO DEFINITIONS
                        ;
                        ; MACRO CALLS
                        ;
                                .MCALL  EMSG$R,NTLER$,NHWDF$,PDVDF$,SLTDF$,NTLDF$,EMSG$
                                .MCALL  SAVRG,RESRG

                        ;
                        ; CONSTANTS AND OFFSETS
                        ;
     000000                     PDVDF$                          ;DEFINE PDV OFFSETS
     000000                     SLTDF$                          ;DEFINE SLT OFFSETS
     000000                     NHWDF$                          ;DEFINE HARDWARE OFFSETS
     000000                     NTLDF$                          ;DEFINE NTL CONTROL BLOCK OFFSETS
```

Table of contents

```
     5-   54     MACRO DEFINITIONS
     6-   84     LOCAL DATA
     7-  159     ERROR MESSAGES
     8-  259     NETWORK INITIALIZATION COLD START
     9-  334     MAIN CONTROL
    10-  469     VECSET - SET UP PROCESS VECTOR TABLE
    11-  539     NTL COMMUNICATION
    12-  568     FIND PDV INDICES OF NSP PROCESSES (ECL AND XPT)
    13-  613     ERROR MESSAGE PROCESSOR
    15-  747     RCV DATA AST
    16-  772     RCV DATA SUBROUTINE
    17-  831     PUTAST - PUT COMPLETION AST IN CALLER TASK'S AST QUEUE
```

```
                MOV     (SP)+,R0        ; COPY UNMAPPED ADDRESS FOR $DEACX

                MOV     (R5),-(SP)      ; SAVE NEXT BLOCK ADDRESS
                MOV     #V.LEN,R1       ; GET VECTOR BLOCK LENGTH
                CALLX   $DEACX,AUX      ; DEALLOCATE BLOCK
                MOV     (SP)+,R5        ; GET NEXT BLOCK
                BNE     10$

        15$:    RESMAP                  ; RESTORE MAPPING
        20$:    RETURN

                .ENDC

```

```
                                                                    H 15
NTINIT     CREATED BY  MACRO  ON 5-SEP-85 AT 11:04       PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE                REFERENCES

ABTIM      000640 RG          #7-224
ALOCB      000620 RG          #7-216       17-859
BUFUMP   = 172354             #5-77
CALLX      000654 RG          #7-235
CCBGT      000664 RG          #7-239
CEACC      000672 RG          #7-242
CF.DDM   = 000002             #5-74
CF.DYN   = 000004             #5-74
CF.EIS   = 000010             #5-74
CF.FRK   = 100000             #5-74
CF.LOG   = 000020             #5-74
CF.MDM   = 000001             #5-74
CF.TIM   = 000400             #5-74
CLKHD      000634 RG          #7-222
CMODE    = 140000             #5-77
DEACB      000622 RG          #7-217
DECPT      000700 RG          #7-245
DEVHD      000642 RG          #7-225
EXEVEC     000606 R           #7-202       7-252
FE.CEX   = ******   GX         9-365
FMASK      000610 RG          #7-203       9-365
FNDPDV     000450 RG          #12-585
HEADR      000626 RG          #7-219
IE.ACT   = ******   GX         6-125
IO.WVB   = ******   GX         13-689
I$$AS    = ******              5-72        5-72         5-75         5-77
KISAR0   = 172340             #5-77
KISAR6   = 172354             #5-77        7-204
KSAR6      000612 RG          #7-204
LDPWF      000644 RG          #7-226
LF.ACT   = 100000             #5-76
LF.BRO   = 000400             #5-76
LF.BWT   = 000007             #5-76
LF.ENA   = 002000             #5-76
LF.LPB   = 001000             #5-76
LF.MDC   = 000100             #5-76
LF.MFL   = 004000             #5-76
LF.MTP   = 000020             #5-76
LF.PAC   = 000200             #5-76
LF.RDY   = 040000             #5-76
LF.REA   = 010000             #5-76
LF.SER   = 000040             #5-76
LF.TIM   = 000010             #5-76
LF.UNL   = 020000             #5-76
LF.X2P   = 000000             #5-76
LGSTT      000676 RG          #7-244       9-399        9-423
LLCAL      000706 RG          #7-248
LLCRS      000666 RG          #7-240
LLCTA      000710 RG          #7-249
LN.CLO   = 000000             #5-76
LN.DUM   = 000005             #5-76
LN.LOA   = 000004             #5-76
```

```
                                                                    I 15
NTINIT     CREATED BY  MACRO  ON 5-SEP-85 AT 11:04       PAGE 2
```

```
                                    .SBTTL  T1SIN - INSERT 1 SECOND TIMER FORK BLOCK
                                    .SBTTL  T50IN - INSERT 50 MSEC TIMER FORK BLOCK
                            ;+
                            ; *** - T1SIN - INSERT 1 SECOND TIMER FORK BLOCK
                            ; *** - T50IN - INSERT 50 MSEC TIMER FORK BLOCK
                            ;
                            ; THIS ROUTINE IS USED TO INSERT ENTRIES ON THE CLOCK QUEUE, WITHOUT CHANGING
                            ; THE PREDEFINED APR5 VALUE STORED IN THE CLOCK BLOCK. THIS ROUTINE IS ONLY
                            ; NEEDED ON MAPPED SYSTEMS.
                            ;
                            ; THIS ROUTINE SHOULD BE CALLED AT SYSTEM STATE - TO INHIBIT TASK SWITCHING
                            ;
                            ; INPUTS:
                            ;       NONE.
                            ;
                            ; OUTPUTS:
                            ;       SPECIFIED CLOCK BLOCK IN INSERT IN CLOCK QUEUE
                            ;
                            ;-
                                    .ENABL  LSB
000116                      T1SIN:
000116  016700  000000G             MOV     T1SCL,R0                ;* GET ADDRESS OF CLOCK QUEUE EN
000122  017702  000000G             MOV     @TKPS,R2                ;* GET # OF CLOCK TICKS PER SECO
000126  000407                      BR      10$                     ;* JOIN COMMON CODE
000130                      T50IN:
000130  005777  000000G             TST     @T100Q                  ;* ANYTHING IN THE 50 MSEC TIMER QUEUE ?
000134  001411                      BEQ     20$                     ;* IF EQ, NO .. DON'T START THE TIMER
000136  016700  000000G             MOV     T100C,R0                ;* ELSE, GET ADDRESS OF CLOCK QUEUE ENTRY
000142  017702  000000G             MOV     @TK100,R2               ;* GET # OF CLOCK TICKS IN 50 MSEC
000146  005001              10$:    CLR     R1                      ;* NO HIGH ORDER # OF TICKS
000150  012704  000006              MOV     #6,R4                   ;* SET SYSTEM REQUEST INDICATOR
000154                              CALLR   CLINS                   ;* INSERT CLOCK BLOCK IN QUEUE
000160                      20$:    RETURN
                                    .DSABL  LSB
```

```
                                          .SBTTL  .DEA16 - DEALLOCATE 16 BIT
                                  ;+
                                  ; *** - .DEA16 - 16 BIT DE-ALLOCATION ROUTINE
                                  ;
                                  ; THIS ROUTINE IS CALLED TO ALLOW BOTH NORMAL AND PARTITION ALLOCATIONS TO
                                  ; BE RELEASED. IF THE ALLOCATION CAME FROM THE PROCESS' PARTITION SPACE, THE
                                  ; LOCATION "..PDV" WILL CONTAIN THE PDV ADDRESS FOR THE PROCESS DE-ALLOCATION
                                  ; OR ZERO IF THE ALLOCATION IS TO BE DISCARDED (I.E., THE PROCESS PARTITION
                                  ; IS ABOUT TO BE RELEASED SO THE DE-ALLOCATION NEED NOT BE HOUSEKEPT).
                                  ;
                                  ; INPUT:
                                  ;       R0 = ALLOCATION ADDRESS
                                  ;       R1 = ALLOCATION SIZE
                                  ;       ..PDV = PDV ADDRESS FOR THE PROCESS OR ZERO
                                  ;
                                  ; OUTPUT:
                                  ;       R0,R1,R2 = DESTROYED
                                  ;       R3,R4,R5 = PRESERVED
                                  ;
                                  ;-

006144                            .DEA16:
006144  010346                            MOV     R3,-(SP)          ; SAVE R3
006146  010446                            MOV     R4,-(SP)          ; SAVE R4

006150  022700  120000                    CMP     #120000,R0        ; IS ALLOCATION FROM PARTITION SPACE ?
006154  101034                            BHI     30$               ; IF HI, NO - DE-ALLOCATE NORMALLY
006156  016702  000202'                   MOV     ..PDV,R2          ; GET PDV ADDRESS
006162  001433                            BEQ     40$               ; IF EQ, DISCARD DE-ALLOCATION
006164  010204                            MOV     R2,R4             ; COPY PDV ADDRESS
006166  016202  000000G                   MOV     Z.PCB(R2),R2      ; GET PCB ADDRESS FOR THIS PROCESS
006172                                    GETRV   R2,.PLGTH         ; READ IN THE PCB
006210  062704  000000G                   ADD     #Z.AVL,R4         ; POINT TO PROCESS ALLOCATION WORD IN PDV
006214  011467  000000G                   MOV     (R4),$PRAVL       ; SETUP FREE SPACE POINTER
006220  042767  160000  000000G           BIC     #160000,$PRAVL    ; IGNORE MAPPING REGISTER SELECT BITS
006226  016767  000014G 000000G           MOV     $BFR+P.REL,$BIAS  ; GET THE BIAS FOR THIS PROCESS
006234                                    CALL    .DEAPR            ; DE-ALLOCATE BACK TO PARTITION SPACE
                                  ;
                                  ; UPDATE THE PROCESS ALLOCATION POINTER IN THE PDV
                                  ;
006240  016714  000000G                   MOV     $PRAVL,(R4)       ; RESTORE PROCESS ALLOCATION POINTER
006244  000402                            BR      40$               ; AND LEAVE

006246                            30$:    CALL    $DEA16            ; DE-ALLOCATE NORMALLY
006252  012604                    40$:    MOV     (SP)+,R4          ; RESTORE R4
006254  012603                            MOV     (SP)+,R3          ; RESTORE R3
006256                                    RETURN
```

I 2

```
VUNL       CREATED BY  MACRO  ON 16-SEP-85 AT 07:31      PAGE 9
MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME        REFERENCES

SWTCHI            #5-59       18-1089    18-1165    21-1508    21-1510    21-1537    21-1539    22-1596    22-1641
SWTCHO            #5-59       17-1047    20-1225    20-1227    23-1773    23-1778
TCBDF$            #5-61       5-78
VECDF$            #5-62       5-89
VNPBPT            #8-261      8-262     #12-557     12-558    #21-1254    21-1255
XPDDB$            #5-61       5-86
```

J 2

```
**FILE**ID**VUNPSI
```

```
    365 001234                                  PUTRC   R5,#2           ; WRITE UPDATED LISTHEAD
    366 001252                                  RESMAP                  ; RESTORE PREVIOUS MAPPING
    367 001256                          50$:    RETURN
    368
```

```
                        ;**
                        ; LIBRARY MACROS
                        ;**
                                .MCALL  ASR$,EMSG$,EMSG$R,NTLER$,SWTCHO,PUTRC,PUTAD,GETRV,GETAD
                                .MCALL  SAVRG,RESRG,CALLR,SWTCHI

                        ;**
                        ; LOCAL DATA
                        ;**

 000000                         .PSECT  DATA,D

                        ;
                        ; BUFFER FOR WRITING VECTOR TO LOW CORE
                        ;
 000000                 VCTBUF: .BLKW   20              ; ALLOW ROOM FOR UP TO 8 VECTOR PAIRS
 000040                 KRBB:   .BLKB   60              ; INTERNAL KRB BUFFER
 000000                         .PSECT
```

```
                                     ;+
                                     ;       **-$VXIT
                                     ;
                                     ;       IF THE USER TYPES ^Z AT TI: THEN A NORMAL EXIT IS TAKEN
                                     ;       BY JUMPING TO $VXIT. OPEN FILES ARE CLOSED AND ANY UNUSUAL
                                     ;       CONDITIONS ARE REPORTED TO THE USER. THINGS ARE NOT ALWAYS
                                     ;       SO NEAT AND CLEAN. A PANIC EXIT CAN BE TAKEN ANYWHERE,
                                     ;       ANYTIME BY JUMPING TO $PANIC.
                                     ;
                                     ; INPUTS:
                                     ;       NONE
                                     ;
                                     ; OUTPUTS:
                                     ;       EXIT$S WITH CONDITION MESSAGES TO TI:
                                     ;-

                                             .MCALL  EXIT$S,CLOSE$,GETRC,GETRV,PUTRC,SWTCHO,SWTCHI,PUTAD
                                             .MCALL  ERMSG$,ERBLK$,ERRPT$,DHBDF$


 000000                                      DHBDF$                          ; DEFINE HOME BLOCK OFFSETS

                                             .ENABL  LC


 000000                              $VXIT::
 000000  005001                              CLR     R1                      ; CLEAR FATAL ERROR FLAG TO PRVENT LOOPS

                                     ; FIRST BE SURE FILE IS OPEN

 000002  005767  000000C                     TST     SYSFDB+F.FNB+N.FID      ; IS FILE ALREADY CLOSED ?
 000006  001002                              BNE     13$                     ; IF NO - BRANCH
 000010                                      CALLR   50$                     ; IF YES - JMP 50$

 000014  032767  000000G 000000G     13$:    BIT     #VF.INI,$VFLAG          ; DID INITIALIZATION COMPLELTE ?
 000022  001002                              BNE     14$                     ; IF YES - BRANCH
 000024                                      CALLR   10$                     ; IF NO - JMP 10$

                                     ; REWRITE RSX11M.SYS VALUES THAT ARE STORED IN ROOT SECTION

 000030                              14$:
                                             .IF DF  R$$MPL                                                  ;[MP01]
                                             SWTCHO  #$CTLST                 ; UPDATE $CTLST                 ;[MP01]
                                             PUTRC   .CTLST,#2

                                             BIT     #DF.POL,$DFLAG          ; SECONDARY POOL SUPPORTED ?    ;[MP01]
                                             BEQ     1$                      ; IF NO - BRANCH                ;[MP01]
                                             SWTCHO  #$POLHD                 ; RESET BUFFER                  ;[MP01]
                                             PUTRC   .POLHD,#2               ; UPDATE IMAGE                  ;[MP01]
                                             .ENDC                                                           ;[MP01]

 000030                              1$:     SWTCHO  #$FMASK                 ; UPDATE $FMASK
 000036                                      PUTRC   .FMASK,#2
 000056                                      SWTCHO  #$DEVHD                 ; UPDATE $DEVHD
 000064                                      PUTRC   .DEVHD
 000076                                      SWTCHO  #$PARHD                 ; UPDATE $PARHD
 000104                                      PUTRC   .PARHD
```

```
                                                                     I  6
VXLB       CREATED BY  MACRO  ON 6-SEP-85 AT 14:39      PAGE 1
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE              REFERENCES

LB.ADR  = ******  GX        5-80
LB.SIZ  = ******  GX        5-81
LB.USE  = ******  GX       *5-75         *5-77
R$$MPL  = ******            5-110         6-130
$DEA18  = ******  GX        5-93          5-114
$GETLB  = ******  GX        5-71          5-101
$NTLHB  = ******  GX        5-95         *5-98
$NTLPT  = ******  GX        5-83
$PUTLB  = ******  GX        5-105
$XLBR     000000 RG        #5-68
.CXLBR  = ******  GX        5-95         *5-98
.MGMGE  = ******  GX        5-73          5-90
```

```
                                                                     J  6
VXLB       CREATED BY  MACRO  ON 6-SEP-85 AT 14:39      PAGE 2
```

```
                                        .SBTTL  DATA


                                ;
                                ; AT THE MOMENT, ASSUME THAT IF MEMORY MANAGEMENT IS IN EFFECT, THAT
                                ; THE MACHINE IS AN 11/70 (ONLY AFFECTS UMR LOADING CODE).
                                ;

                                ;
                                ; LIBRARY MACROS
                                ;
                                        .MCALL  DIR$,CLEF$S,WTSE$S,RQST$,SDAT$,EMSG$R,NTLER$
                                        .MCALL  NHWDF$,PDVDF$,DHBDF$,SLTDF$,CCBDF$

                                        .IF DF  R$$MPL                                          ;[MP01]
                                        .MCALL OLRDF$,QIOW$S,ALUN$S                             ;[MP01]
                                        OLRDF$                                                  ;[MP01]
                                        .ENDC                                                   ;[MP01]

 000000                                 CCBDF$                          ;DEFINE CCB OFFSETS
 000000                                 SLTDF$                          ;DEFINE SLT OFFSETS
 000000                                 NHWDF$                          ;HARDWARE REGISTER DEFINITIONS
 000000                                 PDVDF$                          ;DEFINE PDV OFFSETS
 000000                                 DHBDF$                          ;DEFINE DECNET HOME BLOCK OFFSETS

         000000                 M$$EXT == 0                             ; ASSUME UMR SUPPORT

                                ;
                                ; LOCAL (IMPURE) DATA
                                ;
 000000                                 .PSECT  DATA,D

                                ;
                                ; ERROR MESSAGES
                                ;
                                        .IF     NDF,R$$11S
                                        .ENABL  LC
 000000                         NTLER$  ,10,NONE,$NIERR,,,<Clearing System>
                                        .DSABL  LC
                                        .ENDC

 000030  000030                 INIFUN: .WORD   FS.INI!FC.PCT           ; PROCESS INIT FUNCTION CODE

                                        .IF DF  R$$MPL                                          ;[MP01]
                                STATUS: .BLKW   2                       ; I/O STATUS FOR HRC... INIT    ;[MP01]
                                NTLER$  ,11,NONE,$NIERR,,,<HRC... RECONFIGURATION FAILURE>      ;[MP01]
                                        .ENDC                                                   ;[MP01]

                                ;
                                ; START OF PURE CODE
                                ;
 000000                                 .PSECT
```

Table of contents

```
     5-   54     DATA
     6-   96     CSR CHECK
```

```
     54                                         .SBTTL  MACROS
     55
     56
     57                                 ;
     58                                 ; LIBRARY MACROS
     59                                 ;
     60                                         .MCALL  DIR$,RUN$,EMSG$,EMSG$R,NTLER$,NHWDF$,RETC,WSIG$S
     61                                         .MCALL  ALUN$,PLBDF$,XPDDB$,ECDDB$,PDVDF$,CTRDF$,MAP,BIAS
     62                                         .MCALL  RESMAP,SAVMAP,ADJDF$,NSFDF$,CALLX,CACDF$,DHBDF$
     63                                         .MCALL  DVBDF$,DCBDF$,UCBDF$,HDRDF$,CCBDF$
     64
     65 000000                                  CCBDF$                          ;DEFINE FUNCTION CODE SYMBOLS
     66 000000                                  NHWDF$                          ;HARDWARE REGISTER DEFINITIONS
     67 000000                                  PLBDF$                          ;DEFINE PLB OFFSETS
     68 000000                                  XPDDB$                          ;DEFINE TRANSPORT DATABASE OFFSETS
     69 000000                                  ECDDB$                          ;DEFINE ECL DATABASE OFFSETS
     70 000000                                  PDVDF$                          ;DEFINE PDV TABLE OFFSETS
     71 000000                                  CTRDF$                          ;DEFINE NSP COUNTER OFFSETS
     72 000000                                  NSFDF$                          ;DEFINE NETWORK FLAGS
     73 000000                                  ADJDF$                          ;DEFINE ADJACENCY DATA BASE OFFSETS
     74 000000                                  CACDF$                          ;DEFINE NI CACHE DATA BASE OFFSETS
     75 000000                                  DHBDF$                          ;DEFINE DECNET HOME BLOCK OFFSETS
     76 000000                                  DVBDF$                          ;VCB DEFINITIONS
     77 000000                                  DCBDF$                          ;DCB DEFINITIONS
     78 000000                                  UCBDF$                          ;UCB DEFINITIONS
     79 000000                                  HDRDF$                          ;TASK HEADER OFFSETS
```

```
                                                                 I 10
NTILON     CREATED BY  MACRO  ON 29-JUN-85 AT 03:26       PAGE 2
SYMBOL CROSS REFERENCE                                    CREF    04.00
SYMBOL  VALUE            REFERENCES

PMODE   = 030000         #5-66
PRO     = 000000         #5-66
PR1     = 000040         #5-66
PR2     = 000100         #5-66
PR3     = 000140         #5-66
PR4     = 000200         #5-66
PR5     = 000240         #5-66
PR6     = 000300         #5-66
PR7     = 000340         #5-66
PS      = 177776         #5-66
P$LEN     000052          8-205      8-223
RCP1NM    000004 R       #6-101      8-177
RCP2NM    000010 R       #6-102      8-184
R$$MPL  = ******          5-66       5-66       5-66       8-163
R$$11D  = ******          5-66       5-70
R$$11M  = 000000          5-70
R$$11S  = ******          5-70       6-96       8-198      8-283      9-313
R.UNTN  = 000002          6-104
SRSTD   = ******   GX     8-178      8-185
SWR     = 177570         #5-66
S$$BAS  = ******          7-112      7-112      7-113      7-113      7-114      7-114      7-115      7-115      7-116
                          7-116

TPS     = 177564         #5-66
T$FLAG    000044         #5-71
T$LIF     000013         #5-71
T$LIFL    000013         #5-71
T$LIFO    000013         #5-71
T$LIFS    000013         #5-71
T$LIN     000000         #5-71
T$LIPS    000006         #5-71
T$LLD     000012         #5-71
T$LLDC    000045         #5-71
T$LLDL    000012         #5-71
T$LLDO    000012         #5-71
T$LLDS    000012         #5-71
T$LLEN    000046         #5-71
T$LOPR    000002         #5-71
T$LTCL    000024         #5-71
T$LTIM    000026         #5-71
T$LTPR    000014         #5-71
T$LTPS    000020         #5-71
T$NAPL    000004         #5-71
T$NFE     000000         #5-71
T$NLEN    000010         #5-71
T$NNUL    000002         #5-71
T$NOPL    000006         #5-71
T$NRNI    000042         #5-71
T$NRPL    000005         #5-71
T$NRUL    000007         #5-71
T$NVR     000001         #5-71
T$RPRI    000040         #5-71
T$SVC     000034         #5-71
```

```
                                                                 J 10
NTILON     CREATED BY  MACRO  ON 29-JUN-85 AT 03:26       PAGE 3
```

```
    001200  001366                                BNE     40$                 ;; BR IF NO, TRY NEXT ONE
    001202  126267  000003  176572                CMPB    S$STA(R2),STANUM;;  IS THIS THE CORRECT ONE?
    001210  001362                                BNE     40$                 ;; BR IF NO, TRY NEXT ONE
    001212  017703  000000G                       MOV     @KSAR6,R3           ;; GET APR BIAS OF SVC DESCRIPTOR BLOCK
    001216                                45$:    RESMAP                      ;; RESTORE PREVIOUS MAPPING
    001222  010360  000034                        MOV     R3,T$SVC(R0)        ;; STORE APR BIAS OF SVC DESCRIPTOR BLOCK
    001226  010260  000036                        MOV     R2,T$SVC+2(R0) ;; STORE VIRTUAL ADDRESS OF SVC BLOCK
    001232                                50$:    RESMAP                      ;; RESTORE PREVIOUS MAPPING
    001236                                60$:    RETURN


                                                  .ENDC

            000001                                .END
```

```
NTIMFL  MACRO V05.03b Saturday 29-Jun-85 03:27  Page 6
LOCAL DATA

     69                                                .SBTTL  LOCAL DATA
     70                                        ;
     71                                        ; LOCAL DATA (IMPURE)
     72                                        ;
     73 000000                                         .PSECT  DATA,D
     74 000000  000000                         PDVFLG: .WORD   0                       ;PDV FLAGS FOR DDM
     75
     76                                        ;
     77                                        ; ERROR MESSAGES
     78                                        ;
     79                                                .ENABL  LC
     80 000002                                 NTLER$  ,30,NONE,$NIERR,$NERDD,,<Failed To Load Line * >
     81 000042                                 NTLER$  ,31,NONE,$NIERR,$NER50,,<Failed To Load Process * >
     82                                                .DSABL  LC
     83
     84                                        ;
     85                                        ; START OF PURE CODE
     86                                        ;
     87 000000                                         .PSECT
```

```
NTIMFL  MACRO V05.03b Saturday 29-Jun-85 03:27  Page 7
MARKED FOR LOAD SCAN
```

```
        .TITLE  NTINIT
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       NETWORK INITIALIZER (NTINIT) MAIN CONTROL
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
    539                                         .SBTTL  NTL COMMUNICATION
    540
    541                                 ;+
    542                                 ; $NINTL - SEND DATA TO NTL AND WAIT FOR RESPONSE
    543                                 ;
    544                                 ; THIS ROUINE IS ONLY NECESSARY ON AN RSX11M OR IAS SYSTEM
    545                                 ;
    546                                 ; INPUTS:
    547                                 ;       $NTLFC IS ALREADY SET FOR NTL
    548                                 ;
    549                                 ; OUTPUTS:
    550                                 ;       CARRY INDICATES SUCCESS/FAILURE
    551                                 ;       $NTLFC CONTAINS RESPONSE FROM NTL
    552                                 ;-
    553
    554                                         .IF NDF R$$11S                  ;*-RSX11M SYSTEM ONLY-*
    555
    556 000376                          $NINTL::
    557 000376                                  DIR$    #$NISNL                 ;SEND DATA TO NTL
    558 000404  103416                          BCS     10$                     ;BR IF CAN'T, PROBABLY NOT INSTALLED
    559 000406                                  DIR$    #$NIRNL                 ;REQUEST NTL TO RUN IF NOT ALREADY
    560 000414                                  WTSE$S  #15.                    ;WAIT FOR NTL'S RESPONSE
    561 000426                                  CLEF$S  #15.                    ;CLEAR FLAG 15. FOR NEXT TIME
    562 000440                                  RETURN                          ; WITH C-BIT CLEAR
    563
    564 000442                          10$:    EMSG$R  02                      ;NTL PROBABLY NOT INSTALLED
    565
    566                                         .ENDC                           ;*-END RSX11M CODE-*
```

```
SYMBOL CROSS REFERENCE                                       CREF    04.00

SYMBOL   VALUE              REFERENCES

LN.LOO  = 000003            #5-76
LN.OAU  = 000003            #5-76
LN.OFF  = 000001            #5-76
LN.ON   = 000000            #5-76
LN.OOP  = 000004            #5-76
LN.OPE  = 000001            #5-76
LN.REF  = 000002            #5-76
LN.SER  = 000002            #5-76
LN.STA  = 000017            #5-76
LN.SUB  = 000360            #5-76
LN.TRI  = 000006            #5-76
LO.CON    000006   G        #5-72
LO.CSR    000006   G        #5-72
LO.CTL    000014   G        #5-72
LO.LIN    000004   G        #5-72
LO.PAR    000010   G        #5-72
LO.PRI    000010   G        #5-72
LO.SCR    000014   G        #5-72
LO.STA    000015   G        #5-72
LO.VCT    000012   G        #5-72
LR.AST    000026   G        #5-72        8-305
LR.CIR    000002   G        #5-72
LR.CTL    000004   G        #5-72
LR.EFN    000004   G        #5-72
LR.LIN    000002   G        #5-72
LR.PRM    000030   G        #5-72        8-306
LR.PRO    000002   G        #5-72
LR.STS    000000   G        #5-72
LR.TCB    000002   G        #5-72
LR.UNT    000005   G        #5-72
LS.AST  = 100000   G        #5-72        8-303
LS.CEX  = 000001   G        #5-72        9-361
LS.CIR  = 000010   G        #5-72
LS.CXO  = 010000   G        #5-72        9-389
LS.CX5  = 000400   G        #5-72        9-393        9-415
LS.DLM  = 004000   G        #5-72
LS.ECH  = 001000   G        #5-72
LS.FDX  = 004000   G        #5-72
LS.HDX  = 010000   G        #5-72
LS.LIN  = 000003   G        #5-72
LS.LMC  = 000007   G        #5-72
LS.NTI  = 004000   G        #5-72
LS.ON   = 000011   G        #5-72        9-367
LS.OPT  = 000400   G        #5-72
LS.PRO  = 000002   G        #5-72
LS.PWF  = 040000   G        #5-72
LS.TOP  = 002000   G        #5-72
LS.UNF  = 020000   G        #5-72
LS.X25  = 000012   G        #5-72        9-369
LS.11D  = 004000   G        #5-72
LX.CEX  = 000004   G        #5-72
LX.LIN  = 000006   G        #5-72
```

```
    156                                                .SBTTL  CLINS - INSERT ENTRY ON CLOCK QUEUE
    157                                        ;+
    158                                        ; *** - CLINS - INSERT ENTRY ON CLOCK QUEUE WITHOUT CHANGING APR5 MAPPING VALUE
    159                                        ;
    160                                        ; THIS ROUTINE SHOULD BE CALLED AT SYSTEM STATE - TO INHIBIT TASK SWITCHING
    161                                        ;
    162                                        ; INPUTS:
    163                                        ;       R0 = ADDRESS OF THE CLOCK QUEUE ENTRY BLOCK
    164                                        ;       R1 = HIGH ORDER HALF OF DELTA TIME
    165                                        ;       R2 = LOW ORDER HALF OF DELTA TIME
    166                                        ;       R4 = REQUEST TYPE
    167                                        ;
    168                                        ; OUTPUTS:
    169                                        ;       THE CLOCK QUEUE ENTRY IS INSERTED IN THE CLOCK QUEUE ACCORDING
    170                                        ;       TO THE TIME THAT IT WILL COME DUE.
    171                                        ;
    172                                        ;-
    173 000162                                CLINS:
    174 000162  005020                                CLR     (R0)+                           ;* POINT TO TYPE FIELD AND CLEAR LINK
    175 000164  110420                                MOVB    R4,(R0)+                        ;* SET REQUEST TYPE
    176 000166  005200                                INC     R0                              ;* ADVANCE TO TCB ADDRESS
    177 000170  005720                                TST     (R0)+                           ;* SKIP TCB LOCATION
    178 000172  067702  000000G                       ADD     @ABTIM,R2                       ;* CALCULATE ABSOLUTE TIME WHEN
    179 000176  005501                                ADC     R1                              ;* REQUEST WILL COME DUE
    180 000200  010220                                MOV     R2,(R0)+                        ;* SET LOW ORDER PART OF TIME
    181 000202  010110                                MOV     R1,(R0)                         ;* SET HIGH ORDER PART OF TIME
    182 000204  016702  000000G                       MOV     CLKHD,R2                        ;* POINT TO CLOCK QUEUE LIST HEAD
    183 000210  010201                        10$:    MOV     R2,R1                           ;* SAVE ADDRESS OF PREVIOUS ENTRY
    184 000212  011102                                MOV     (R1),R2                         ;* GET ADDRESS OF NEXT ENTRY
    185 000214  001410                                BEQ     20$                             ;* IF EQ, END OF LIST
    186 000216  021062  000010                        CMP     (R0),C.TIM+2(R2)                ;* ENTRY GO HERE ?
    187 000222  101372                                BHI     10$                             ;* IF HI, NO
    188 000224  103404                                BLO     20$                             ;* IF LO, ENTRY GOES HERE
    189 000226  026062  177776  000006                CMP     -2(R0),C.TIM(R2)                ;* REALLY GO HERE ?
    190 000234  103365                                BHIS    10$                             ;* IF HIS, NO
    191 000236  162700  000010                20$:    SUB     #8.,R0                          ;* BACK UP TO FRONT OF BLOCK
    192 000242  010011                                MOV     R0,(R1)                         ;* SET NEW NEXT IN PREVIOUS ENTRY
    193 000244  010210                                MOV     R2,(R0)                         ;* SET NEXT IN NEW ENTRY
    194 000246                                        RETURN
    195
    196
```

```
                                                .SBTTL  TSKSCH - SEARCH FOR TASK TCB
                                        ;+
                                        ; **-TSKSCH-SEARCH TCB'S FOR SPECIFIED TASK
                                        ;
                                        ; INPUTS:
                                        ;       R0      - POINTER TO RAD50 TASK NAME
                                        ;
                                        ; OUTPUTS:
                                        ;       $RADDR  - ADDRESS OF TCB
                                        ;       $BFR    - IMAGE OF TCB
                                        ;       C-BIT SUCCESS/FAILURE
                                        ;       R3,R4,R5 DESTROYED
                                        ;
                                        ;-

006260                                  TSKSCH:
006260                                          GETRV   .TSKHD,.TLGTH   ; READ IN LISTHEAD
006300  012705  000000G                         MOV     #$BFR,R5        ; POINT AT BUFFER BASE
006304  011565  000030                          MOV     (R5),T.TCBL(R5) ; INITIALIZE FOR LOOP

006310  016567  000030  000000G 10$:            MOV     T.TCBL(R5),$RADDR ; SET TCB ADDRESS
006316  000261                                  SEC                     ; ASSUME ERROR
006320  001411                                  BEQ     30$             ; NULL TASK ( END OF LIST )
006322                                          GETRV                   ; READ IN TCB
006326  021065  000006                          CMP     (R0),T.NAM(R5)  ; MUST MATCH BOTH
006332  001366                                  BNE     10$
006334  026065  000002  000010                  CMP     2(R0),T.NAM+2(R5) ; HALVES OF NAME
006342  001362                                  BNE     10$             ; C-BIT IS CLEAR
006344                          30$:            RETURN

        000001                                  .END
```

```
**FILE**ID**VUNPSI
```

```
VV      VV   UU      UU   NN      NN   PPPPPPPP       SSSSSSSS     IIIIII
VV      VV   UU      UU   NN      NN   PPPPPPPP       SSSSSSSS     IIIIII
VV      VV   UU      UU   NN      NN   PP      PP   SS               II
VV      VV   UU      UU   NN      NN   PP      PP   SS               II
VV      VV   UU      UU   NNNN    NN   PP      PP   SS               II
VV      VV   UU      UU   NNNN    NN   PP      PP   SS               II
VV      VV   UU      UU   NN  NN  NN   PPPPPPPP       SSSSSS         II
VV      VV   UU      UU   NN  NN  NN   PPPPPPPP       SSSSSS         II
VV      VV   UU      UU   NN    NNNN   PP                   SS       II
VV      VV   UU      UU   NN    NNNN   PP                   SS       II
  VV  VV     UU      UU   NN      NN   PP                   SS       II           ....
  VV  VV     UU      UU   NN      NN   PP                   SS       II           ....
    VV       UUUUUUUUUU   NN      NN   PP           SSSSSSSS       IIIIII         ....
    VV       UUUUUUUUUU   NN      NN   PP           SSSSSSSS       IIIIII         ....


LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
VUNPSI - VNP UNLOAD ROUTINES FO MACRO V05.03b Wednesday 17-Jul-85 12:14
Table of contents
```

```
                                                    .SBTTL  XPORT - DEALLOCATE PORT TABLE AND CIRCUIT BLOCKS

                                            ;+
                                            ;
                                            ; XPORT - DEALLOCATE PORT TABLE AND ASSOCIATED X.25 CIRCUIT BLOCKS
                                            ;
                                            ; INPUTS:
                                            ;       $PFLAG - PSI characteristics word                               ;[TD001]
                                            ;                                                                       ;**-1
                                            ; OUTPUTS:
                                            ;       THE PORT TABLE AND THE X.25 CIRCUIT BLOCK IS DEALLOCATED
                                            ;
                                            ;
                                            ;-
001260  016703  000000G                     XPORT:  MOV     $PSIPT,R3           ; GET ADDRESS OF PSI HOME BLOCK
001264  062703  000020                              ADD     #H$PTB,R3           ; POINT TO PORT TABLE ADDRESS
001270                                              GETRV   R3,#2               ; READ IN PORT TABLE ADDRESS
001306  016700  000000G                             MOV     $BFR,R0             ; GET ADDRESS OF PORT TABLE
001312  062703  000002                              ADD     #H$NPT-H$PTB,R3     ; POINT TO NUMBER OF PORTS
001316                                              GETRV   R3,#2               ; READ IN NUMBER OF PORTS
001334  016704  000000G                             MOV     $BFR,R4             ; GET NUMBER OF PORTS

001340                                              CALL    $SAVAL              ; SAVE ALL REGISTERS
001344                                              CEACC$  R0,R2               ; CONVERT TO MAPPED ADDRESS
001354  010405                                      MOV     R4,R5               ; GET NUMBER OF PORTS
001356  006305                                      ASL     R5                  ; GET NUMBER OF BYTES IN PORT TABLE
001360                                              GETAD   R2,R5               ; READ IN PORT TABLE
001374  012702  000000G                             MOV     #$BFR,R2            ; POINT TO LOCAL COPY OF PORT TABLE
001400  005722                                      TST     (R2)+               ; SKIP FIRST ENTRY
001402  005304                                      DEC     R4                  ; ...
001404  003424                                      BLE     25$                 ; IF NO CIRCUIT BLOCKS - BRANCH
001406  012200                              10$:    MOV     (R2)+,R0            ; GET ADDRESS OF X.25 CIRCUIT BLOCK
001410  001420                                      BEQ     20$                 ; BR IF NONE FOR THIS PORT
001412  012701  000106                              MOV     #X$LEN,R1           ; GET LENGTH OF CIRCUIT BLOCK
001416  032767  000000G 000000G                     BIT     #PF.PSV,$PFLAG      ; Check for PSV module              ;[TD001]
001424  001402                                      BEQ     15$                 ; Branch if not present             ;[TD001]
001426  012701  000106                              MOV     #X$GLEN,R1          ; PSV needs longer XCB              ;[TD001]
001432                                      15$:    SAVMAP                      ; SAVE PORT TABLE MAPPING           ;[TD001]
001436                                              SAVRG   <R2>                ; AND ADDRESS                       ;**-1
001440                                              CALL    $DEPOL              ; DEALLOCATE CIRCUIT BLOCK
001444                                              RESRG   <R2>                ; RESTORE PORT TABLE ADDRESS
001446                                              RESMAP                      ; AND MAPPING
001452  005304                              20$:    DEC     R4                  ; MORE TO CHECK?
001454  003354                                      BGT     10$                 ; BR IF YES
001456  016600  000002                      25$:    MOV     R$R0(SP),R0         ; GET UNMAPPED ADDRESS OF PORT TABLE
001462  016601  000012                              MOV     R$R4(SP),R1         ; GET NUMBER OF PORTS
001466  006301                                      ASL     R1                  ; MULTIPLY BY 2 TO OBTAIN BYTE COUNT
001470                                              CALL    $DEPOL              ; DEALLOCATE PORT TABLE
001474                                      30$:    RETURN

```

```
                                ;+
                                ; $TMVCT - SETUP INTERRUPT VECTOR(S)
                                ;
                                ; INPUTS:
                                ;       $$VECT=VECTOR TO BE USED
                                ;       $NVECT=NUMBER OF VECTORS TO SET UP
                                ;       $ICBAD=ICB ADDRESS/LINE TABLE ADDRESS
                                ;       $VECT1=INTERRUPT DISPLACEMENT VALUE(S)
                                ;
                                ; OUTPUTS:
                                ;       R4 - preserved
                                ;-

000000                          $TMVCT::SAVRG   <R4>            ; Save reg
000002  005767  000000G                 TST     $FLAG1          ; ANY ERRORS YET?
000006  100453                          BMI     15$             ; IF MI, YES - DO NOTHING

000010  016701  000000G                 MOV     $PDVA,R1        ; GET PDV ADDRESS
000014  032761  000000G 000000G         BIT     #ZF.PSE,Z.FLG(R1) ; IS THIS A PSEUDO-DDM?
000022  001045                          BNE     15$             ; BR IF YES
000024  032767  000000G 000000G         BIT     #FL.DDM,$FLAGS  ; ARE WE LOADING A DDM?
000032  001441                          BEQ     15$             ; IF EQ, NO - DO NOTHING
000034  032767  000000G 000000G         BIT     #FL.CHA,$FLAGS  ; MUX CHARACTERISTICS UPDATE ONLY ?
000042  001035                          BNE     15$             ; YES .. DO NOTHING

000044  012700  000000'                 MOV     #VCTBUF,R0      ; GET FCS BUFFER ADDRESS
000050  012701  000000G                 MOV     #$VECT1,R1      ; GET VECTOR DISPLACEMENT(S) ADDRESS
000054  016702  000000G                 MOV     $NVECT,R2       ; GET NUMBER OF VECTORS
000060  010203                          MOV     R2,R3           ; TWICE
000062  006303                          ASL     R3              ; CONVERT TO A BYTE COUNT
000064  006303                          ASL     R3              ;       ...

                                        .IF DF  R$$MPL
                                        MOV     .MPRO,R4        ; GET NUMBER OF PROCESSORS               ;[MP01]
                                        BPL     17$             ; IF MULTIPROCESSOR - BRANCH             ;[MP01]
                                        ADD     #2,$ICBAD       ; SKIP LINK WORD                         ;[MP01]
                                        .ENDC

000066  016710  000000G         10$:    MOV     $ICBAD,(R0)     ; COPY ICB/LINE TABLE ADDRESS
000072  062120                          ADD     (R1)+,(R0)+     ; ADD APPROPRIATE DISPLACEMENT
000074  012720  000000G                 MOV     #PR7,(R0)+      ; SET PRIORITY 7
000100  005302                          DEC     R2
000102  001371                          BNE     10$             ; LOOP $NVECT TIMES

000104                                  SWTCHO  #VCTBUF         ; POINT AT OUTPUT BUFFER

                                        .IF NDF R$$MPL
000112                                  PUTRC   $$VECT,R3       ; RE-WRITE VECTOR(S)
                                        .IFF
                                        TST     .MPRO           ; MULTIPROCESSOR ?                       ;[MP01]
                                        BLT     12$             ; IF NO - BRANCH                         ;[MP01]
                                        PUTAD   $$VECT,R3       ; RE-WRITE VECTORS (MULTIPROCESSOR)      ;[MP01]
                                        BR      14$             ; CONTINUE                               ;[MP01]
                                12$:    PUTRC   $$VECT,R3       ; RE-WRITE VECTORS                       ;[MP01]
                                        .ENDC

000130                          14$:    SWTCHO                  ; POINT BACK AT INPUT BUFFER
```

```
112 000116                                      SWTCHO  #$CRAVL                 ; UPDATE $CRAVL
113 000124                                      PUTRC   .CRAVL                  ; REWRITE RSX POOL LISTHEAD
114
115                                             .IF DF  R$$MPL                                                  ;[MP01]
116
117                                             BIT     #DF.DAS,$DFLAG          ; D-SPACE ENABLED ?             ;[MP01]
118                                             BEQ     4$                      ; IF NO - BRANCH                ;[MP01]
119
120                                             SWTCHO  #$ICAVL                 ; REWRITE ICB POOL              ;[MP01]
121                                             PUTRC   .ICAVL                  ; ...                           ;[MP01]
122
123                                             MOV     .MPRO,R3                ; MULTIPROCESSOR SYSTEM ?       ;[MP01]
124                                             BLT     4$                      ; IF NO - BRANCH                ;[MP01]
125                                     2$:     DEC     R3                      ; GET NEXT CPU NUMBER           ;[MP01]
126                                             BEQ     4$                      ; IF NO MORE - BRANCH           ;[MP01]
127                                             MOV     R3,R1                   ; COPY CPU NUMBER               ;[MP01]
128
129                                             ASL     R1                      ; MULTIPLY BY 8.                ;[MP01]
130                                             ASL     R1                      ; ...                           ;[MP01]
131                                             ASL     R1                      ; ...                           ;[MP01]
132                                             ADD     #$ICAVL,R1              ; GET ADDRESS OF INTERNAL LISTHEAD
133                                             MOV     R3,R2                   ; COPY CPU NUMBER               ;[MP01]
134                                             ASL     R2                      ; CREATE WORD INDEX             ;[MP01]
135
136                                             ADD     .K6TAB,R2               ; CALCULATE CPU BIAS ADDRESS    ;[MP01]
137                                             SWTCHI  #$CPBIA                 ; SET BUFFER ADDRESS            ;[MP01]
138                                             GETRV   R2,#2                   ; READ BIAS                     ;[MP01]
139                                             MOV     $CPBIA,$RBLCK           ; SET UP CPU BIAS               ;[MP01]
140                                             SWTCHO  R1                      ; SET BUFFER ADDRESS            ;[MP01]
141                                             PUTAD   .ICAVL,#4               ; UPDATE LISTHEAD               ;[MP01]
142                                             BR      2$                      ; CONTINUE                      ;[MP01]
143                                             .ENDC                                                           ;[MP01]
144
145 000136                              4$:     SWTCHI                          ; RESET BUFFERS                 ;[MP01]
146 000144                                      SWTCHO
147
148 000152  005767  000000G                     TST     $DECPT                  ; DECNET HOME BLOCK ALLOCATED ? ;[MP01]
149 000156  001424                              BEQ     5$                      ; IF NO - BRANCH                ;[MP01]
150 000160                                      GETRV   $DECPT,#D$END           ; READ DECNET HOME BLOCK        ;[MP01]
151 000200  122767  000040  000006G             CMPB    #40,$BFR+D$LNAM         ; BLANK NODE NAME ?             ;[MP01]
152 000206  001010                              BNE     5$                      ; IF NO - BRANCH                ;[MP01]
153
154 000210  016700  000000G                     MOV     $DECPT,R0               ; IF YES - PSI ONLY SYSTEM      ;[MP01]
155 000214  012701  000104                      MOV     #D$END,R1               ; GET HOME BLOCK SIZE           ;[MP01]
156 000220                                      CALL    $DEA16                  ; DEALLOCATE                    ;[MP01]
157 000224  005067  000000G                     CLR     $DECPT                  ; INIT HOME BLOCK POINTER
158
159                                     ;       REWRITE COMEXEC IMAGES IF LOADED
160
161 000230  032767  000000G 000000G 5$:     BIT     #FE.CEX,$FMASK          ; IS COMEXEC LOADED ?
162 000236  001402                              BEQ     10$                     ; NO
163 000240                                      CALL    $VIMO                   ; YES, THEN REWRITE CORE IMAGES TO DISK
164                                     ;
165                                     ; REWRITE TRANSFER ADDRESS INDICATING SUCCESS
166                                     ;
167 000244  032767  000000G 000000G 10$:    BIT     #V1.NSY,$VFLG1          ; DID WE FIND A SYSTEM IMAGE FILE?
168 000252  001017                              BNE     20$                     ; BR IF NO
```

```
VXLB       CREATED BY  MACRO  ON 6-SEP-85 AT 14:39      PAGE 2
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              5-71        5-93        5-101       5-105       5-114
CEACC$           #4-53
GETAD            #4-53
RETURN            5-115
VECDF$           #4-53        4-55
```

```
**FILE**ID**VXPR
```

```
                                                .SBTTL  SET CEX FUNCTION

                                  ;+
                                  ; $NICEX - SET CEX FUNCTION ENTRY POINT
                                  ;
                                  ; THE SAME ENTRY POINT GLOBAL IS DEFINED FOR BOTH RSX11M AND RSX11S
                                  ; SYSTEMS.
                                  ;
                                  ; INPUTS:
                                  ;       ON AN RSX11M SYSTEM ONLY, FE.CEX IN "$FMASK" INDICATES
                                  ;         WHETHER CEX IS ALREADY LOADED (1) OR NOT (0).
                                  ;       ON AN RSX11S SYSTEM, CEX IS ASSUMED TO BE ALREADY
                                  ;         LOADED.
                                  ;       LS.CXO IN "$NIFNC" INDICATES WHETHER THE FUNCTION IS
                                  ;         LOAD CEX ONLY (0) OR ALSO PERFORM LOCAL ON (1).
                                  ;
                                  ; OUTPUTS:
                                  ;       CEX IS LOADED, UMR'S ARE LOADED, CSR'S ARE CHECKED
                                  ;         AND THE TIMERS ARE STARTED, AS NECESSARY.
                                  ;       ON RSX11M SYSTEMS WHERE CEX IS NOT YET LOADED, NTL IS
                                  ;         CALLED TO PERFORM INITIALIZATION, AND A "MARKED FOR
                                  ;         LOAD" AND "MARKED FOR ENABLE" SCAN IS PERFORMED.
                                  ;       ON RSX11M SYSTEMS WHERE CEX IS ALREADY LOADED, LOCAL
                                  ;         INITIALIZATION IS PERFORMED.
                                  ;       ON AN RSX11S SYSTEM, LOCAL INITIALIZATION ONLY IS
                                  ;         PERFORMED (FALL THROUGH TO LOCAL INIT CODE).
                                  ;       CARRY IS SET IF LOAD OR INITIALIZATION FAILED.
                                  ;
                                  ;-

000000                            $NICEX::.IF   NDF,R$$11S                  ;*-THE FOLLOWING IS RSX11M CODE-*

000000  032777  000000G 000000G         BIT     #FE.CEX,@FMASK              ;IS CEX LOADED ALREADY?
000006  001040                          BNE     NTILIN                      ;BR IF YES, PERFORM LOCAL INIT
000010  012767  000000C 000000G         MOV     #LS.CEX+LS.NTI,$NTLFC       ;SET NTL FUNCTION CODE
000016                                  CALL    $NINTL                      ;SEND DATA TO NTL
000022  103421                          BCS     20$                         ;BR ON ERROR (NO NTL)
000024  005767  000000G                 TST     $NTLFC                      ;WHAT WAS NTL'S RESPONSE
000030  000261                          SEC                                 ; ASSUME FAILURE
000032  100415                          BMI     20$                         ;BR IF BAD RESPONSE
000034  032767  000000G 000000G         BIT     #LS.TOP,$NTLFC              ; COPY TOP-LOADING FLAG
000042  001402                          BEQ     10$                         ; FROM NTL'S ANSWER
000044  105267  000001G                 INCB    $NIFLG+1                    ; TO HIGH BYTE OF $NIFLG
000050                            10$:  CALL    FNDPDV                      ;FIND NSP PROCESSES (ECL & XPT)
000054                                  CALL    $NIMFL                      ;PERFORM MARKED FOR LOAD SCAN
000060  103403                          BCS     30$                         ;BR IF LOAD FAIL, REMOVE CEX
000062                                  CALL    $NITIM                      ;START THE TIMERS IF NECESSARY
000066                            20$:  RETURN                              ;RETURN TO MAIN LINE CODE

000070  012767  000000G 000000G   30$:  MOV     #LX.CEX,$NTLFC              ;IF LOAD ERROR, REMOVE CEX
000076                                  CALL    $NINTL                      ;LET NTL DO IT
000102                                  EMSG$R  10                          ;TELL USER AND RETURN W/ CARRY SET

                                        .ENDC                               ;*-END OF RSX11M CODE-*
```

```
                        .TITLE  NTICSR
                        .IDENT  /V05.00/
                ;
                ;
                ; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                ;
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                ; TRANSFERRED.
                ;
                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                ; CORPORATION.
                ;
                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                ;
                ; MODULE DESCRIPTION:
                ;
                ;       NTINIT - SCAN CSR LIST AND TEST DEVICES FOR ONLINE
                ;
                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                ;
                ; IDENT HISTORY:
                ;
                ; 1.00  27-FEB-78
                ;       VERSION 2.0 RELEASE
                ;
                ; 2.00  14-DEC-79
                ;       DECNET-11M/S V3.0
                ;       DECNET-11M-PLUS V1.0
                ;
                ; 3.00  16-APR-82
                ;       DECNET-11M V3.1
                ;       DECNET-11M-PLUS V1.1
                ;
                ; 4.00  07-NOV-83
                ;       DECNET-11M V4.0
                ;       DECNET-11M-PLUS V2.0
                ;
                ; 5.00  22-JUL-85
                ;       DECnet-11M/S V4.2
                ;       DECnet-11M-Plus V3.0
                ;       DECnet-Micro/RSX V1.0
                ;
                ;

```

```
                                                 .SBTTL  LOCAL DATA

                                         ;
                                         ; CONSTANT DEFINITIONS
                                         ;
                176000                   AREAMK  = 176000                  ; AREA BITS MASK
                002000                   LVL1    = 2000                    ; LEVEL 1 AREA

                                         ; LOCAL DATA (IMPURE)
                                         ;
        000000                                   .PSECT  DATA,D

                                         ;
                                         ; LOCAL STORAGE
                                         ;
                                                 .IF NDF R$$11S
        000000                           PLDADR: .BLKW   1                 ; PLD DATA BASE ADDRESS
        000002                           PLDLEN: .BLKW   1                 ; LENGTH OF THE PLD DATA BASE
                                                 .ENDC   ; NDF R$$11S

        000004  070410  143074           RCP1NM: .RAD50  /RCP1../          ; ROUTING CONTROL PROGRAM (LEVEL 1)
        000010  070410  146174           RCP2NM: .RAD50  /RCP2../          ; ROUTING CONTROL PROGRAM (LEVEL 2)

        000014                           LONACP: RUN$    NETACP,,,,,1,1    ;DIRECTIVE TO REQUEST NETACP
        000042                           ASSIGN: ALUN$   2,NS,0            ;ASSIGN LUN 2 TO NS:
```

```
                                                      J 10
NTILON     CREATED BY  MACRO  ON 29-JUN-85 AT 03:26     PAGE 3
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE              REFERENCES

T$T5      000030           #5-71
T$T6      000032           #5-71
UBMPR   = 170200           #5-66
UISAR0  = 177640           #5-66
UISAR1  = 177642           #5-66
U.CW3     000014            8-174
X$$MCB  = ******            5-70       5-70
ZF.COU  = 001000           #5-70
ZF.DDM  = 000001           #5-70
ZF.DIA  = 004000           #5-70
ZF.DLC  = 000002           #5-70
ZF.DVP  = 100000           #5-70
ZF.INI  = 040000           #5-70
ZF.KMX  = 000020           #5-70
ZF.LLC  = 000004           #5-70
ZF.LMC  = 000100           #5-70
ZF.MAN  = 020000           #5-70
ZF.MFL  = 000010           #5-70
ZF.MTM  = 000400           #5-70
ZF.MUX  = 000040           #5-70
ZF.PSE  = 002000           #5-70
ZF.SLI  = 010000           #5-70
ZF.TIM  = 000200           #5-70
ZF.X3P  = 000000           #5-70
ZS.ASN  = 100000           #5-70
ZS.BSY  = 140000           #5-70
Z.AVL     000014           #5-70
Z.DAT     000016           #5-70
Z.DSP     000000           #5-70       5-70
Z.FLG     000010           #5-70
Z.LEN   = 000016           #5-70
Z.LLN     000006           #5-70       8-158
Z.MAP     000020           #5-70
Z.NAM     000004           #5-70
Z.PCB     000012           #5-70
Z.SCH     000007           #5-70
$DBECL  = ******   GX       8-144      8-270
$DBXPT  = ******   GX       8-142      8-196
$ERR11    000052 RG        #7-112      8-297
$ERR15    000142 RG        #7-113      8-152      8-192      8-294
$ERR17    000202 RG        #7-114      8-295
$ERR18    000256 R         #7-115      8-231      8-296
$ERR19    000316 RG        #7-116      8-195
$HEADR  = ******   GX       9-320
$MUL    = ******   GX       8-206
$NIERR  = ******   GX       7-112      7-113      7-114      7-115      7-116
$NILON    000000 RG        #8-140
$NIMFE  = ******   GX       8-276
$NIXPT  = ******   GX       8-154
$$$GLB  = ******            6-104      6-105
$$$T1   = 000002           #6-104      6-104      6-104      6-104     #6-105
$$$T2   = 000046 R         #6-105      6-105
```

```
                                                      K 10
NTILON     CREATED BY  MACRO  ON 29-JUN-85 AT 03:26     PAGE 4
```

```
                                                                 J 11
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:08  Page 8-2
Symbol table

ACTCIR  000004R          CM.XLO= 000004           C.STS   000012           E$NCR   000034           FS.SPW= 040000
AT$ACL= 000100           CP.DCF= 000040           C.URM   177776           E$NCS   000036           FS.STM= 000000
AT$ADN= 000040           CP.HDL= 000007           C.XACP  000004           E$NIC   000044           FS.STP= 002000
AT$AUP= 000020           CP.PS = 177400           C.XID   000035           E$NLEN  000050           FS.STR= 001000
AT$CYC= 000004           CP.PSI= 000200           C.XLEN  000044           E$NLLA  000012           FS.TRM= 003000
AT$LV1= 000002           CP.XCF= 000100           C.XPLI  000040           E$NLNK  000000           FS.WLB= 001000
AT$LV2= 000001           CP.2FR= 000030           C.XPT   000034           E$NML   000040           FS.XKL= 002000
AT$NEX= 000200           CS.ABO= 000100           C.XSVC  000042           E$NMR   000024           FS.XOF= 010000
AT$UP = 000010           CS.BRO= 000002           C.XTC   000037           E$NMS   000030           FS.XON= 007000
A$CIR   000003           CS.BUF= 000200           C.X25   000036           E$NNOD  000002           FS.ZER= 002000
A$LEN   000004           CS.CES= 000002           DECPT = ****** GX        E$NRT   000042           F$$LVL= 000001
A$NID   000000           CS.CHN= 000010           DLMSET  000772R          E$NRTP  000005           G$$TPP= 000000
A$TM    000003           CS.CMP= 000200           D$AMXC  000072           E$NSEG  000010           G$$TSS= 000000
A$TMI   000002           CS.DCR= 000400           D$AMXH  000074           E$NTIM  000046           G$$TTK= 000000
A$TSZ   000000           CS.DEF= 000004           D$ANN   000000           E$NUSE  000004           G$$WRD= 000000
A$TYP   000002           CS.DEV= 000002           D$BRPR  000102           E$STRT  000006           HF$DLM= 000002
A$$CHK= 000000           CS.DIS= 000040           D$BRTM  000100           E$$XPR= 000000           HF$GWY= 000010
A$$CPS= 000000           CS.ENA= 000001           D$DELF  000045           FC.CCP= 000020           HF$HOS= 000004
A$$PRI= 000000           CS.ENB= 000020           D$DELW  000046           FC.CTL= 000006           HF$XDF= 000020
A$$TRP= 000000           CS.ERR= 100000           D$END = 000104           FC.KCP= 000016           H$CUG   000010
CB.CCB= 000002           CS.FTL= 001000           D$FNB   000034           FC.KIL= 000004           H$DST   000012
CB.DDM= 000040           CS.HCR= 000001           D$HIOR  000024           FC.MAN= 000024           H$D29   000014
CB.DLC= 000020           CS.HFE= 002000           D$HOST  000022           FC.MLD= 000026           H$FLG   000000
CB.RDB= 000004           CS.LST= 040000           D$INAC  000044           FC.PCT= 000030           H$GLEN  000104
CB.SDB= 000010           CS.MTL= 004000           D$INCT  000042           FC.PWR= 000022           H$GLT   000044
CB.SLI= 000100           CS.RNG= 000010           D$IPL   000051           FC.RCE= 000002           H$GNAM  000050
CB.XLB= 000001           CS.ROV= 000004           D$LID   000020           FC.RCP= 000014           H$GNML= 000020
CCBGT = ****** GX        CS.RSN= 010000           D$LNAM  000006           FC.TIM= 000010           H$GPT   000046
CC.LLC= 000200           CS.SHU= 000001           D$LNUM  000014           FC.XCP= 000012           H$HITS  000034
CEACC = ****** GX        CS.SID= 000002           D$LST   000047           FC.XME= 000000           H$HLEN  000044
CE.ABO= 100362           CS.STR= 000004           D$MAXC  000064           FNDPDV= ****** GX        H$LBDA  000070
CE.DAO= 100346           CS.SUC= 000001           D$MAXH  000066           FS.AST= 000000           H$LBDN  000072
CE.DIS= 100366           CS.TMO= 020000           D$MAXV  000070           FS.CIB= 002000           H$LDTE  000002
CE.ERR= 100370           CS.XUR= 000004           D$MLL   000040           FS.CRA= 001000           H$LEN   000042
CE.ILN= 100350           CTLNUM  000003R          D$MNOD  000041           FS.DIS= 013000           H$LOTS  000032
CE.LTO= 100356           C$$CKP= 000000           D$NA    000062           FS.DVC= 001000           H$NETW  000024
CE.MOP= 100372           C$$ORE= 000400           D$NBEA  000056           FS.ENB= 012000           H$NML = 000006
CE.NTE= 100361           C$$RSH= 177564           D$NBRA  000054           FS.EXI= 001000           H$NPT   000022
CE.RTE= 100376           C.ADD   000034           D$NEND= 000054           FS.GET= 006000           H$PTB   000020
CE.SRC= 100364           C.BID   000003           D$NLN   000030           FS.HLT= 000000           H$PVC   000006
CE.STP= 100352           C.BUF   000014           D$NN    000060           FS.INI= 000000           H$RDTE  000004
CE.TME= 100354           C.BUF1  000014           D$OUTT  000043           FS.KIL= 000000           H$RNW   000042
CE.TMO= 100374           C.BUF2  000024           D$RETF  000050           FS.LCL= 100000           H$SVC   000036
CE.UNS= 100344           C.CNT   000020           D$RNN   000002           FS.LTM= 001000           H$TRB   000016
CF.CHN= 000001           C.CNT1  000020           D$RTMR  000076           FS.MNT= 004000           H$XAVL  000100
CF.EOM= 000004           C.CNT2  000030           D$SEG   000036           FS.MSN= 014000           H$XBIA  000074
CF.HDR= 000020           C.FLG   000022           D$SER   000032           FS.REA= 001000           H$X29C  000040
CF.LB = 100000           C.FLG1  000022           D$SQRL  000052           FS.RET= 000000           I$$RAR= 000000
CF.LIN= 000002           C.FLG2  000032           D$$BUG= 177514           FS.REZ= 003000           I$$RDN= 000000
CF.SOM= 000010           C.FNC   000010           D$$ISK= 000000           FS.RLB= 002000           KISAR6= ****** GX
CF.SYN= 000040           C.LIN   000006           D$$L11= 000001           FS.RNG= 011000           KSAR6 = ****** GX
CF.TRN= 000100           C.LNK   000000           D$$YNC= 000000           FS.RST= 000000           K$$CNT= 177546
CM.CIR= 000002           C.MOD   000011           D$$YNM= 000000           FS.RTN= 001000           K$$CSR= 177546
CM.FMT= 100000           C.NSP   000004           ENAC    000074R          FS.SET= 005000           K$$LDC= 000000
CM.HRD= 000002           C.PRO   000042           ENALIN  000610R          FS.SFC= 005000           K$$TPS= 000074
CM.LIN= 000000           C.RSV   000002           E$NBR   000014           FS.SFR= 006000           LD$LP = 000000
CM.LOO= 000001           C.STA   000007           E$NBS   000020           FS.SFS= 004000           LF.ACT= 100000
```

```
                                                                 K 11
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:08  Page 8-3
Symbol table
```

```
                                             .SBTTL  MARKED FOR LOAD SCAN

                                   ;+
                                   ; $NIMFL - PERFORM 'MARKED FOR LOAD' SCAN
                                   ;
                                   ; SCAN THE COMMEXEC DATABASE TO DISCOVER WHICH PROCESSES AND
                                   ; LINES HAVE THE 'MARKED FOR LOAD' ATTRIBUTE. REQUEST THE NETWORK
                                   ; LOADER (NTL) TO LOAD ALL LINES AND PROCESSES SO MARKED.
                                   ;
                                   ; INPUTS:
                                   ;         $SLTTA  = CONTAINS ADDRESS OF FIRST SLT
                                   ;         $SLTNM  = CONTAINS NUMBER OF SLT'S
                                   ;         $PDVTA  = CONTAINS ADDRESS OF FIRST PDV
                                   ;         $PDVNM  = CONTAINS NUMBER OF PDV'S
                                   ;         NF.TOP IN "$NIFLG" INDICATES BOTTOM UP (0) OR TOP DOWN (1) LOAD
                                   ;
                                   ; OUTPUTS:
                                   ;         LINES AND PROCESSES ARE LOADED AS REQUIRED.
                                   ;         R0, R1, R2 ARE DESTROYED.
                                   ;-

000000                             $NIMFL::
000000  017701  000000G                      MOV     @PDVTA,R1                    ;POINT TO PDV INDEX TABLE
000004  017700  000000G                      MOV     @PDVNM,R0                    ;GET NUMBER OF ENTRIES
000010  012102                     10$:      MOV     (R1)+,R2                     ;GET A PDV POINTER
000012  001434                               BEQ     30$                          ;BR IF EMPTY ENTRY
000014  132762  000010  000010               BITB    #ZF.MFL,Z.FLG(R2)            ;IS PROCESS MARKED FOR LOAD?
000022  001430                               BEQ     30$                          ;BR IF NO
000024  012767  000002  000000G              MOV     #LS.PRO,$NTLFC+LR.STS        ;SET FUNCTION CODE
000032  105767  000001G                      TSTB    $NIFLG+1                     ;TOP-DOWN ALLOCATION?
000036  001403                               BEQ     20$                          ;BR IF NO
000040  052767  002000  000000G              BIS     #LS.TOP,$NTLFC+LR.STS        ;MARK THE REQUEST BLOCK
000046  016267  000004  000002G    20$:      MOV     Z.NAM(R2),$NTLFC+LR.PRO      ;SET THE PROCESS NAME
000054                                       CALL    $NINTL                       ;REQUEST NTL TO LOAD PROCESS
000060  103530                               BCS     100$                         ;BR IF NTL NOT INSTALLED
000062  005767  000000G                      TST     $NTLFC                       ; WAS NTL SUCCESSFUL ?
000066  100006                               BPL     30$                          ; IF PL, YES
000070  016767  000002G 000000G              MOV     $NTLFC+LR.PRO,$NIERA         ;SET THE PROCESS NAME
000076                                       EMSG$R  31                           ;FAILED TO LOAD PROCESS ''PRO''
                                                                                  ; NO RETURN
000104                             30$:      SOB     R0,10$                       ;LOOP TILL ALL PDV'S HAVE BEEN SCANNED
```

```
     54                                         .SBTTL  MACRO DEFINITIONS
     55
     56                                 ;
     57                                 ; MACRO CALLS
     58                                 ;
     59                                         .MCALL  EMSG$,NTLER$,QIOW$S,EXIT$S,ASTX$S,WTSE$S,WSIG$S,CLEF$S
     60                                         .MCALL  DIR$,CLEF$S,SETF$S,SRDA$S,RCVD$,SDAT$,RQST$,DSAR$S,ENAR$S
     61                                         .MCALL  CCBDF$,NHWDF$,NTLDF$,PDVDF$,SLTDF$,OPTDF$,MAP,CALLX,CALLR
     62                                         .MCALL  SDAT$S,EMSG$R,RCVX$,RETC,MRKT$,VECDF$
     63                                         .MCALL  SCBDF$,PCBDF$,TCBDF$,SAVMAP,RESMAP
     64
     65                                         .IF DF  R$$MPL
     66                                         .MCALL  GIN$
     67                                         .ENDC
     68
     69                                 ;
     70                                 ; CONSTANTS AND OFFSETS
     71                                 ;
     72 000000                                  NTLDF$  <:>,<=>,NTLSYM          ;NTL REQ. BLOCK AND SYMBOLS (GLOBALLY)
     73 000000                                  CCBDF$                          ;CCB OFFSETS
     74 000000                                  OPTDF$                          ;COMM/EXEC OPTIONS WORD
     75 000000                                  PDVDF$                          ;PDV OFFSETS
     76 000000                                  SLTDF$                          ;SLT OFFSETS
     77 000000                                  NHWDF$                          ;HARDWARE OFFSETS
     78 000000                                  PCBDF$                          ;PARTITION CONTROL BLOCK OFFSETS
     79 000000                                  SCBDF$                          ;UMR BLOCK O.D
     80 000000                                  TCBDF$                          ;TCB OFFSETS
     81 000000                                  VECDF$                          ;VECTOR BLOCK OFFSETS
     82
```

```
                                                    .SBTTL  FIND PDV INDICES OF NSP PROCESSES (ECL AND XPT)
                                          ;+
                                          ; FNDPDV - FIND PDV INDICES OF NSP PROCESSES (ECL AND XPT)
                                          ;
                                          ; SCAN THE PDV INDEX TABLE FOR PROCESSES WITH THE NAMES 'ECL' AND 'XPT'
                                          ; AND SAVE THEIR RESPECTIVE PDV INDICES AND DATA BASE ADDRESSES.
                                          ;
                                          ; INPUTS:
                                          ;       NONE
                                          ;
                                          ; OUTPUTS:
                                          ;       $NIECL = ECL PDV INDEX OR ZERO
                                          ;       $DBECL = ADDRESS OF ECL'S DDB OR ZERO
                                          ;       $NIXPT = XPT PDV INDEX OR ZERO
                                          ;       $DBXPT = ADDRESS OF XPT'S DDB OR ZERO
                                          ;-

000450  005067  000000'                   FNDPDV::CLR     $NIECL                  ;ASSUME ECL AND XPT
000454  005067  000002'                           CLR     $NIXPT                  ;  ARE NOT IN THE SYSTEM
000460  005067  000004'                           CLR     $DBECL                  ;   ...
000464  005067  000006'                           CLR     $DBXPT                  ;   ...

000470                                            SWSTK$  20$                     ; ENTER SYSTEM STATE
000474  012702  017704                            MOV     #^RECL,R2               ;;LOOK FOR ECL'S PDV
000500                                            CALL    @PDVID                  ;; ...
000504  103413                                    BCS     10$                     ;;IF CS, NOT FOUND
000506  010267  000000'                           MOV     R2,$NIECL               ;;SAVE ECL'S PDV INDEX
000512  067702  000712'                           ADD     @PDVTA,R2               ;;POINT INTO PDV INDEX TABLE
000516  011202                                    MOV     (R2),R2                 ;;GET ADDRESS OF ECL'S PDV
000520  005762  000012                            TST     Z.PCB(R2)               ;;IS THE PROCESS LOADED?
000524  001403                                    BEQ     10$                     ;;IF EQ, NO
000526  016267  000016  000004'                   MOV     Z.DAT(R2),$DBECL        ;;SAVE ADDRESS OF ECL'S DATABASE

000534  012702  114224                    10$:    MOV     #^RXPT,R2               ;;LOOK FOR XPT'S PDV
000540                                            CALL    @PDVID                  ;; ...
000544  103413                                    BCS     20$                     ;;IF CS, NOT FOUND
000546  010267  000002'                           MOV     R2,$NIXPT               ;;SAVE XPT'S PDV INDEX
000552  067702  000712'                           ADD     @PDVTA,R2               ;;POINT INTO PDV INDEX TABLE
000556  011202                                    MOV     (R2),R2                 ;;GET ADDRESS OF XPT'S PDV
000560  005762  000012                            TST     Z.PCB(R2)               ;;IS THE PROCESS LOADED?
000564  001403                                    BEQ     20$                     ;;IF EQ, NO
000566  016267  000016  000006'                   MOV     Z.DAT(R2),$DBXPT        ;;SAVE ADDRESS OF XPT'S DATABASE

000574                                    20$:    RETURN
```

```
SYMBOL CROSS REFERENCE                                       CREF    04.00

SYMBOL   VALUE               REFERENCES

LX.PRO  = 000005  G          #5-72
L.COST    000015             #5-76
L.CTL     000012             #5-76
L.CVA     177776             #5-76
L.DDM     000002             #5-76
L.DDS     000004             #5-76
L.DLC     000003             #5-76
L.DLM     000006             #5-76
L.DLS     000010             #5-76
L.FLG     000000             #5-76
L.KRBA    000016             #5-76
L.LEN   = 000022             #5-76
L.MPF     000022             #5-76
L.NMST    000020             #5-76
L.NSTA    000014             #5-76
L.OWNR    000021             #5-76
L.UNT     000013             #5-76
MPAR    = 172100             #5-77
MPCSR   = 177746             #5-77
MRKTEF  = 000001             #6-108        6-109        9-402        9-426
M$$MGE  = 000000              5-77
M.KTEF  = 000002              6-109
NTIBSC    000224 R            6-124       #6-125
NTIBSY    000212 R           #6-124       *16-811      *16-812       16-813
NTIERB    000243 R           #7-165        13-669
NTIERN    000230 R           #7-164        13-688       13-689
NTIGRQ    000060 R           #9-355        9-449
NTINIT    000000 R           #8-284        17-881
NTIRCV    000124 R           #6-116        15-763
NTIRCX    000016 R           #6-104        16-796
NTIRDA    001042 R            8-300       #15-762
NTIRDT    001100 R            15-765      #16-798
NTIRDX    001072 R            8-299        9-447       #16-796
NTISEN    000172 RG          #6-122        9-446       *16-820      *16-821       17-854
NTLPT     000652 RG          #7-234
N$$VCT  = ******              17-862
PDVID     000704 RG          #7-247        12-592       12-602
PDVNM     000702 RG          #7-246
PDVTA     000712 RG          #7-250        12-595       12-605
PIRQ    = 177772             #5-77
PMODE   = 030000             #5-77
PR0     = 000000             #5-77
PR1     = 000040             #5-77
PR2     = 000100             #5-77
PR3     = 000140             #5-77
PR4     = 000200             #5-77
PR5     = 000240             #5-77
PR6     = 000300             #5-77
PR7     = 000340             #5-77
PS      = 177776             #5-77
PSIPT     000670 RG          #7-241
PUMR      000624 RG          #7-218
```

```
                                     .IF DF   R$$MPL                  ; VNP'D SYSTEM IMAGE HACK       ;[MP01]

                             ;+
                             ; THIS ROUTINE WILL BE REMOVED FOR NEXT BASE LEVEL,AFTER A
                             ; MORE GENERAL SOLUTION IS DESIGNED .
                             ;
                             ;
                             ; $NXHCK - INIT DLX THRU POWERFAIL CALL IF SYSTEM IS VNP'D
                             ;
                             ; INPUTS: NONE
                             ;
                             ; OUTPUTS: DLX CALLED AT POWERFAIL SYSTEM IS VNP'D
                             ;+

                             $NXHCK::CMP      NTISEN+S.DATN,#^RNTL ; IS THIS FROM NTL ?
                                     BEQ      20$                 ; IF YES - BRANCH  (NX POWERFAIL COMPLETED)

                                     SWSTK$   20$                 ; ENTER KERNEL STATE

                                     MOV      #''NX,R3            ;* GET DEVICE NAME
                                     MOV      @DEVHD,R0           ;* GET HEAD OF DEVICE LIST
                             10$:    MOV      (R0),R0             ;* GET NEXT DEVICE
                                     BEQ      20$                 ;* IF END - BRANCH
                                     CMP      R3,D.NAM(R0)        ;* IS THIS NX ?
                                     BNE      10$                 ;* IF NO - BRANCH

                                     MOV      R0,R3               ;* SAVE DCB ADDRESS
                                     MOV      D.UCB(R3),R5        ;* GET UCB ADDRESS
                                     MOV      U.SCB(R5),R4        ;* GET SCB ADDRESS
                                     MOV      D.DSP(R3),R2        ;* GET DISPATCH TABLE
                                     CALL     @LDPWF              ;* CALL NX AT POWERFAIL ROUTINE

                             20$:    RETURN


                                     .ENDC


        000001                       .END
```

```
                                                               K 1
VUNL - VNP UNLOAD ROUTINES      MACRO V05.03b Monday 16-Sep-85 07:30  Page 25-1
Symbol table

AB    = ****** GX       D$DELF  000045          E$NTIM  000046          K.PRI   000000          N$GTM   000015
AS.DEL= 000010          D$DELW  000046          E$NUSF  000004          K.VCT   000001          N$HC1   000052
AS.EXT= 000004          D$END = 000104          E$STRT  000006          LD$LP = 000000          N$HC2   000056
AS.RED= 000001          D$FNB   000034          E$$XPR= 000000          LF.ACT= ****** GX       N$HIGH  000033
AS.WRT= 000002          D$HIOR  000024          FATAL = ****** GX       LF.ENA= ****** GX       N$LLT   000026
AT$ACL= 000100          D$HOST  000022          FE.CEX= ****** GX       LF.RDY= ****** GX       N$LLTM  000024
AT$ADN= 000040          D$INAC  000044          FF.ADD= 000020          LF.UNL= ****** GX       N$LVC   000036
AT$AUP= 000020          D$INCT  000042          FF.CIR= 000040          L$$ASG= 000000          N$LV1   000002
AT$CYC= 000004          D$IPL   000051          FF.CON= 000001          L$$DRV= 000000          N$LV2   000010
AT$LV1= 000002          D$LID   000020          FF.FIL= 000002          L$$P11= 000001          N$MBXQ  000050
AT$LV2= 000001          D$LNAM  000006          FF.HST= 040000          L$$11R= 000000          N$MHC1  000036
AT$NEX= 000200          D$LNUM  000014          FF.LIN= 000010          L.CTL = ****** GX       N$MHC2  000044
AT$UP = 000010          D$LST   000047          FF.MOD= 000100          L.DDM = ****** GX       N$PLD   000016
A$CIR   000003          D$MAXC  000064          FF.MDN= 000004          L.DDS = ****** GX       N$PLLT  000030
A$LEN   000004          D$MAXH  000066          FF.MSK= 000077          L.DLC = ****** GX       N$PRI   000076
A$NID   000000          D$MAXV  000070          FF.PRT= 000200          L.DLM = ****** GX       N$ROA1  000022
A$TM    000003          D$MLL   000040          FF.QUL= 000370          L.DLS = ****** GX       N$ROA2  000030
A$TMI   000002          D$MNOD  000041          FF.REM= 100000          L.FLG = ****** GX       N$RTMX  000014
A$TSZ   000000          D$NA    000062          FLT     005242R         L.KRBA= ****** GX       N$RTM1  000014
A$TYP   000002          D$NBEA  000056          FLT16   005242R         M$$CRB= 000124          N$RTM2  000015
A$$CHK= 000000          D$NBRA  000054          FL.PX1= ****** GX       M$$CRX= 000000          N$RT1   000000
A$$CPS= 000000          D$NEND= 000054          FL.PX2= ****** GX       M$$FCS= 000000          N$RT2   000006
A$$PRI= 000000          D$NLN   000030          FNDADD  002174R         M$$MGE= 000000          N$SLA   000016
A$$TRP= 000000          D$NN    000060          F$$LVL= 000001          M$$NET= 000000          N$SNOD  000012
A.ACC   000020          D$OUTT  000043          F.ADD   000016          M$$OVR= 000000          N$TCTL  000112
A.IOC   000003          D$RETF  000050          F.CEV   000030          M.BFVH  000011          N$TIM   000004
A.LGTH= 000014          D$RNN   000002          F.CIR   000024          M.BFVL  000012          N$TLC   000100
A.LNK   000000          D$RTMR  000076          F.CLS   000002          M.LGTH= 000014          N$TNC   000114
A.MPCT  000011          D$SEG   000036          F.EVT   000004          M.LNK   000000          N$TRC   000106
A.NAM   000002          D$SER   000032          F.FLG   000014          M.UMRA  000002          N$TTCB  000110
A.PCB   000012          D$SQRL  000052          F.LEN   000016          M.UMRN  000004          N$VCB   000010
A.PCBL  000000          D$$BUG= 177514          F.LIN   000020          M.UMVH  000010          N$VER   000066
A.PRI   000002          D$$ISK= 000000          F.LNK   000000          M.UMVL  000006          N$XLEN  000124
A.PUD   000010          D$$L11= 000001          F.MOD   000022          NF$BLK= 000100          N$$ACC= 000001
A.REM   000012          D$$YNC= 000000          F.NRBD= ****** GX       NF$DMO= 000010          N$$BUF= 000001
A.STAT  000010          D$$YNM= 000000          F.REM   000026          NF$MOU= 000040          N$$LDV= 000001
A.TCB   000004          D.DSP = ****** GX       F.RSIZ= ****** GX       NF$RST= 000002          N$$MCP= 000001
A.TCBL  000006          D.NAM = ****** GX       F.SEV   000004          NF$SCN= 000020          N$$MLL= 000001
A.UCB   000010          D.PCB = ****** GX       F.URBD= ****** GX       NF$SHU= 000004          N$$MOV= 000010
BLOCK   000012R     002 ERRA    001606R         GETIND  000522R         NF$TIM= 000200          N$$NCT= 000001
CLQREM= ****** GX       ERR44   003126R         GETPDV  002150R         NSITAB  000130R     002 N$$PEM= 000001
COUNT   000116R     002 ERR77   003664R         G$$TPP= 000000          NTIR    000124R     002 OF.PRO= 000040
CSRUNL  003440R         E$NBR   000014          G$$TSS= 000000          N$ACQ   000000          OF.RLU= 000100
C$$CKP= 000000          E$NBS   000020          G$$TTK= 000000          N$ACTL  000032          OF.SMC= 000200
C$$ORE= 000400          E$NCR   000034          G$$WRD= 000000          N$ADJ1  000072          O.FLG   000003
C$$RSH= 177564          E$NCS   000036          ICBLK   000150R     002 N$ADJ2  000074          O.LEN   000012
DDMAC   000122R     002 E$NIC   000044          ICBSAV  000174R     002 N$CACH  000062          O.LNK   000000
DEAECL  001226R         E$NLEN  000050          ICBSC   003132R         N$CIR   000034          O.MXC   000004
DEAXPT  000564R         E$NLLA  000012          ICBSZ   000176R     002 N$CRC   000120          O.NAM   000006
DEVEC   000000R         E$NLNK  000000          I$$RAR= 000000          N$DLA   000020          O.TYP   000002
DLCAC   000120R     002 E$NML   000040          I$$RDN= 000000          N$DLY   000014          PF$BLK= 020000
DSIZE = ****** GX       E$NMR   000024          KMCB    000114R     002 N$ELEN  000054          PF$CLC= 010000
D$AMXC  000072          E$NMS   000030          K$$CNT= 177546          N$ENC   000042          PF$DLM= 100000
D$AMXH  000074          E$NNOD  000002          K$$CSR= 177546          N$ERRC  000022          PF$EIP= 000002
D$ANN   000000          E$NRT   000042          K$$LDC= 000000          N$FLG   000005          PF$ENB= 000001
D$BRPR  000102          E$NRTP  000005          K$$TPS= 000074          N$FNC   000006          PF$FAI= 004000
D$BRTM  000100          E$NSEG  000010          K.CSR   000002          N$GENQ  000052          PF$FM1= 000100
```

```
                                                               L 1
VUNL - VNP UNLOAD ROUTINES      MACRO V05.03b Monday 16-Sep-85 07:30  Page 25-2
Symbol table
```

Table of contents

```
     5-   52     MACRO DEFINITIONS
     6-   86     LOCAL SYMBOL DEFINITIONS
     7-   98     LOCAL DATA
     8-  105     $XPSI - Unload PSI data bases                              ;[TD001]
     9-  158     DEAPVC - Deallocate PVC name blocks                        ;[TD001]
    10-  203     DEACUG - Deallocate CUG blocks                             ;[TD001]
    11-  249     DEADST - Deallocate destination descriptors                ;[TD001]
    12-  308     XLDTE - DEALLOCATE LOCAL DTE BLOCK
    13-  370     XPORT - DEALLOCATE PORT TABLE AND CIRCUIT BLOCKS
    14-  420     XX29 - DEALLOCATE X.29 DATA BLOCK
    15-  446     XTRA - DEALLOCATE TRACE DATA STRUCTURES
    16-  490     XPSV - Deallocate PSV data structures                      ;[TD001]
    17-  522     DEALST - DEALLOCATE LIST OF DATA BLOCKS
    420                                                       .SBTTL  XX29 - DEALLOCATE X.29 DATA BLOCK
    421
    422                                               ;+
    423                                               ; XX29 - DEALLOCATE X.29 DATA BLOCK
    424                                               ;
    425                                               ; INPUTS
    426                                               ;       NONE
    427                                               ;
    428                                               ; OUTPUTS
    429                                               ;       R0, R1, R2 DESTROYED
    430                                               ;       THE X.29 DATA BLOCK IS DEALLOCATED
    431                                               ;
    432                                               ;-
    433
    434 001476  016701  000000G               XX29:   MOV     $PSIPT,R1               ; POINT TO HOME BLOCK
    435 001502  062701  000040                        ADD     #H$X29C,R1              ; POINT TO X.29 BLOCK
    436 001506                                        GETRV   R1,#2                   ; READ IN X.29 BLOCK ADDRESS
    437 001524  016700  000000G                       MOV     $BFR,R0                 ; GET X.29 BLOCK ADDRESS
    438 001530  001410                                BEQ     10$                     ; IF EQ, NOTHING TO DO
    439 001532  005067  000000G                       CLR     $BFR                    ; FORGET POINTER
    440 001536                                        PUTRC                           ; REWRITE UPDATED POINTER
    441 001542  012701  000016                        MOV     #Q$LEN,R1               ; GET SIZE OF BLOCK
    442 001546                                        CALL    $DEA16                  ; DEALLOCATE IT
    443 001552                                10$:    RETURN                          ; AND RETURN
    444
```

```
000136         15$:    RESRG   <R4>                ; Restore reg
000140                 RETURN

                       .IF DF  R$$MPL

               17$:    SAVRG   <R0,R1,R2,R3>       ; SAVE REG'S                                  ;[MP01]
                       MOV     $SLTA,R5            ; GET SLT ADDRESS                             ;[MP01]
                       MOV     L.KRBA(R5),R5       ; GET KRB ADDRESS                             ;[MP01]
                       ASL     R4                  ; CONVERT CPU NUMBER TO WORD INDEX            ;[MP01]
                       SWTCHI  #KRBB               ; SET INPUT BUFFER ADDRESS                    ;[MP01]
                       ADD     #K.PRM,R5           ; BACK OFF ADDRESS TO START OF KRB            ;[MP01]
                       GETRV   R5,#$KRBLN          ; READ KRB                                    ;[MP01]
                       MOV     #KRBB,R5            ; GET INTERNAL ADDRESS OF KRB                 ;[MP01]
                       SUB     #K.PRM,R5           ; ADJUST ADDRESS TO OFFSET 0                  ;[MP01]
               20$:    SUB     #2,R4               ; ADJUST CPU NUMBER                           ;[MP01]
                       BLT     22$                 ; IF NO MORE - BRANCH                         ;[MP01]
                       MOV     .URMTB,R3           ; GET UNIBUS RUN CONTINUITY TABLE ADR.        ;[MP01]
                       ADD     R4,R3               ; POINT TO CORRECT CPU ENTRY                  ;[MP01]
                       SWTCHI  #$BFR               ; SET INPUT BUFFER                            ;[MP01]
                       GETRV   R3,#2               ; READ UNIBUS RUN CONNECTIVITY ENTRY          ;[MP01]
                       BIT     K.URM(R5),$BFR      ; CONTROLLER ON THIS CPU ?                    ;[MP01]
                       BEQ     20$                 ; IF NO - BRANCH                              ;[MP01]
                       BR      25$                 ; IF YES - CONTINUE                           ;[MP01]
               22$:    IOT                         ; DIDN'T FIND UNIBUS RUN                      ;[MP01]

               25$:    ADD     #K.PRM,R5           ; POINT AT ICB ADDRES/2                       ;[MP01]
               27$:    MOV     (R5),R5             ; GET NEXT ICB                                ;[MP01]
                       BEQ     40$                 ; IF NO ICB - BRANCH                          ;[MP01]
                       MOV     R5,R2               ; COPY ICB/2 ADDRESS                          ;[MP01]
                       BIC     #7777,R2            ; GET CPU NUMBER                              ;[MP01]
                       BIC     R2,R5               ; CLEAR CPU NUMBER                            ;[MP01]
                       SWAB    R2                  ; PLACE CPU NUMBER IN LOWER BITS              ;[MP01]
                       ASR     R2                  ; ...                                         ;[MP01]
                       ASR     R2                  ; ...                                         ;[MP01]
                       ASR     R2                  ; ...                                         ;[MP01]
                       ASL     R5                  ; CREATE ACTUAL ICB ADDRESS                   ;[MP01]
                       MOV     R5,$ICBAD           ; COPY ICB ADDRESS                            ;[MP01]
                       MOV     .K6TAB,R3           ; GET CPU PARTION BIAS ADDRESS                ;[MP01]
                       ADD     R2,R3               ; GET CORRECT CPU ENTRY                       ;[MP01]
                       GETRV   R3,#2               ; READ CPU PARTITION BIAS                     ;[MP01]
                       MOV     $BFR,$RBLCK         ; SET UP FOR BIAS ADDRESSING ACCESS           ;[MP01]
                       CMP     R2,R4               ; CORRECT CPU ?                               ;[MP01]
                       BEQ     30$                 ; IF YES - BRANCH                             ;[MP01]
                       GETAD   R5,#2               ; READ FIRST WORD OF NEXT ICB                 ;[MP01]
                       MOV     #$BFR,R5            ; SET UP INTERNAL BUFFER ADDRESS              ;[MP01]
                       BR      27$                 ; CONTINUE                                    ;[MP01]

               30$:    ADD     #2,$ICBAD           ; SKIP PAST LINK POINTER                      ;[MP01]
                       RESRG   <R3,R2,R1,R0>       ; RESTORE REG'S                               ;[MP01]
                       CALLR   10$                 ; CONTINUE                                    ;[MP01]

               40$:    IOT                         ; NO ICB ??                                   ;[MP01]

                       .ENDC

```

```
000254  012767  176000  000000G          MOV     #-<2*512.>,$RADDR       ; SET UP ADDRESS
000262  012767  000002G 000000G          MOV     #L$BBLK+2,$RSIZE        ; REREAD LABEL BLOCK
000270  005067  000000G                  CLR     $RBLCK                  ; NO BIAS
000274                                   GETRC                           ; ...

                                         .IF NDF R$$MPL                                                  ;[MP01]
000300  016767  000000G 000000C          MOV     $XFRAD,$BFR+L$BXFR      ; SET PROPER TRANSFER ADDRESS
                                         .IFF                                                            ;[MP01]
                                         MOV     $XFRSV,$BFR+L$BXFR      ;  ...
                                         .ENDC

000306                                   CALL    $PUTRD                  ; REWRITE TRANSFER ADDRESS
                                 ;
                                 ; CLOSE SYSTEM IMAGE FILE
                                 ;
000312  016767  000000G 000000G  20$:    MOV     $RSIZ,$RSIZE            ; RESTORE ORIGINAL RECORD SIZE

000320                                   CLOSE$  #SYSFDB
000330  103403                           BCS     110$
                                 ;
                                 ; SCRAM
                                 ;
000332                           50$:    EXIT$S

                                 ;
                                 ; ERRORS
                                 ;
000340  012746  000332'          110$:   MOV     #50$,-(SP)      ; REPORT ERROR AND EXIT
000344                                   ERRPT$  ERR1,$EROUT     ; ...

000354                                   ERMSG$  <Failed to close system image file ( %E ).>
000425                                   ERBLK$  ERR1,<SYSFDB+F.ERR>


        000001                           .END
```

```
**FILE**ID**VXPR
```

```
VV      VV    XX      XX    PPPPPPPP     RRRRRRRR
VV      VV    XX      XX    PPPPPPPP     RRRRRRRR
VV      VV    XX      XX    PP      PP   RR      RR
VV      VV    XX      XX    PP      PP   RR      RR
VV      VV      XX  XX      PP      PP   RR      RR
VV      VV      XX  XX      PP      PP   RR      RR
VV      VV        XX        PPPPPPPP     RRRRRRRR
VV      VV        XX        PPPPPPPP     RRRRRRRR
VV      VV      XX  XX      PP           RR  RR
VV      VV      XX  XX      PP           RR  RR
  VV  VV      XX      XX    PP           RR    RR          ....
  VV  VV      XX      XX    PP           RR    RR          ....
    VV        XX      XX    PP           RR      RR        ....
    VV        XX      XX    PP           RR      RR        ....
```

```
LL               SSSSSSSS    TTTTTTTTTT
LL               SSSSSSSS    TTTTTTTTTT
LL             SS                TT
LL             SS                TT
LL             SS                TT
LL             SS                TT
LL               SSSSSS          TT
LL               SSSSSS          TT
LL                     SS        TT
LL                     SS        TT
LL                     SS        TT
LL                     SS        TT
LLLLLLLLLL     SSSSSSSS          TT
LLLLLLLLLL     SSSSSSSS          TT
```

```
                                                .SBTTL  LOCAL INITIALIZATION

                                        ;+
                                        ; NTILIN - LOCAL INITIALIZATION
                                        ;
                                        ; LOAD UMR'S AND VALIDATE CSR'S AS NEEDED
                                        ;
                                        ; THIS ROUTINE REQUIRES THE GLOBAL ADDRESSES "$NIUMR", AND "$NICSR"
                                        ; TO BE DEFINED.  ANY OF THESE ROUTINES MAY BE EXCLUDED FROM
                                        ; THE TASK BUILD IF IT IS KNOWN THAT THEY WILL NOT BE CALLED (IF
                                        ; EXCLUDED, THE SYMBOLS SHOULD BE GLOBALLY DEFINED AS FOLLOWS:
                                        ; $NIUMR=177777, $NICSR=0).  ALSO, ON AN RSX11-S SYSTEM,
                                        ; THE GLOBAL "$NIDLI" SHOULD BE DEFINED (ZERO IF MODULE IS EXCLUDED).
                                        ;
                                        ; INPUTS:
                                        ;       $NTLPT          = ADDRESS (BIAS) OF NTL DATA CONTROL BLOCK
                                        ;       BLOCK+2         = ADDRESS OF UMR CONTROL BLOCK (0 IF NONE)
                                        ;       BLOCK+.CXCSR    = ADDRESS (BIAS) OF CSR CHECK LIST (0 IF NONE)
                                        ;
                                        ; OUTPUTS:
                                        ;       "$NIDLI" IS CALLED TO INIT DOWN-LINE LOAD PARAMETERS (11S ONLY)
                                        ;       IF BLOCK+2      IS NON-ZERO, "$NIUMR" IS CALLED
                                        ;       IF BLOCK+.CXCSR IS NON-ZERO, "$NICSR" IS CALLED
                                        ;       IF AN ERROR OCCURS IN ANY OF THESE, THE INITIALIZATION FAILED
                                        ;       IF THE INITIALIZATION SUCCEEDED, THE TIMERS ARE STARTED
                                        ;       CARRY IS SET IF INITIALIZATION FAILED
                                        ;-

000110                                  NTILIN::CALL    FNDPDV                  ;FIND NSP PROCESSES (ECL & XPT)

000114  017700  000000G                         MOV     @PDVTA,R0               ; GET PDV TABLE ADDRESS
000120  017701  000000G                         MOV     @PDVNM,R1               ; GET NUMBER OF PDV'S
000124  012002                          2$:     MOV     (R0)+,R2                ; GET NEXT PDV
000126  032762  040000  000010                  BIT     #ZF.INI,Z.FLG(R2)       ; PDV NEED INITIALIZATION ?
000134  001426                                  BEQ     4$                      ; IF NO - BRANCH
000136  005762  000012                          TST     Z.PCB(R2)               ; IF PROCESS LOADED ?
000142  001423                                  BEQ     4$                      ; IF NO - BRANCH
000144  016205  000016                          MOV     Z.DAT(R2),R5            ; GET DATA BASE ADDRESS
000150  011503                                  MOV     (R5),R3                 ; GET FIRST WORD
000152  006003                                  ROR     R3                      ; HAS PROCESS BEEN INITIALIZED ?
000154  103415                                  BCS     3$                      ; IF YES - BRANCH

000156                                          SWSTK$  3$                      ; ENTER SYSTEM STATE
000162  016202  000004                          MOV     Z.NAM(R2),R2            ;* GET PDV NAME
000166                                          CALL    @PDVID                  ;* GET PDV INDEX
000172  016746  000030'                         MOV     INIFUN,-(SP)            ;* GET INIT FUNCTION CODE
000176  010603                                  MOV     SP,R3                   ;* GET ADDRESS OF FUNCTION CODE
000200                                          CALL    @LLCAL                  ;* DISPATCH TO PROCESS
000204  005726                                  TST     (SP)+                   ;* CLEAN UP STACK

000206                                          RETURN

000210  000241                          3$:     CLC
000212  005301                          4$:     DEC     R1                      ; COUNT ONE LESS PDV
000214  001343                                  BNE     2$

                                                .IF     DF,R$$11S               ;*-CODE FOR AN RSX11S SYSTEM-*
```

```
     54                                         .SBTTL  DATA
     55
     56                                 ;
     57                                 ; LIBRARY MACROS
     58                                 ;
     59                                         .MCALL  DIR$,SVTK$,EMSG$,NTLER$,NHWDF$,SLTDF$,PDVDF$,SAVRG,RESRG
     60
     61 000000                                  NHWDF$                          ;HARDWARE REGISTER DEFINITIONS
     62 000000                                  SLTDF$                          ;SYSTEM LINE TABLE OFFSETS
     63 000000                                  PDVDF$                          ;PDV OFFSETS
     64
     65
     66                                 ;
     67                                 ; LOCAL (IMPURE) DATA
     68                                 ;
     69 000000                                  .PSECT  DATA,D
     70
     71                                 ;
     72                                 ; DIRECTIVES TO ALLOW PROGRAM TO TRAP ILLEGAL ADDRESS USAGE (FOR CSR CHECK)
     73                                 ;
     74 000000                          CSRENA: SVTK$   CSRTAB,1                ;TRAP USING ''CSRTAB'' TABLE
     75 000006                          CSRDIS: SVTK$                           ;DISABLE INTERNAL PROGRAM TRAPS
     76
     77 000014  000274'                 CSRTAB: .WORD   CSRTRP                  ;TRAP TO ''CSRTRP'' ON ILLEGAL ADDRESS
     78
     79                                 ;
     80                                 ; WORD TO HOLD SYSTEM LINE TABLE ADDRRESS
     81                                 ;
     82 000016                          CSRSLT: .BLKW   1                       ;SYSTEM LINE TABLE ADDRESS
     83
     84                                 ;
     85                                 ; ERROR MESSAGES
     86                                 ;
     87                                         .ENABL  LC
     88 000020                          NTLER$  <:>,20,NONE,$NIERR,$NERDN,,<Device * Offline>
     89                                         .DSABL  LC
     90
     91                                 ;
     92                                 ; PURE CODE SECTION
     93                                 ;
     94 000000                                  .PSECT
```

```
    107                                         .SBTTL  ERROR MESSAGES
    108                                 ;
    109                                 ; ERROR MESSAGES
    110                                 ;
    111                                         .ENABL  LC
    112 000052                          NTLER$  <:>,11,NONE,$NIERR,,,<Router Priorities Data Base Allocation Failure>
    113 000142                          NTLER$  <:>,15,NONE,$NIERR,,,<NS: Assignment Failure>
    114 000202                          NTLER$  <:>,17,NONE,$NIERR,,,<Network ACP (NETACP) Not Installed>
    115 000256                          NTLER$  ,18,NONE,$NIERR,,,<PLB Allocation Failure>
    116 000316                          NTLER$  <:>,19,NONE,$NIERR,,,<Routing Control Task Not Installed>
    117                                         .DSABL  LC
    118
    119                                 ;
    120                                 ; START OF PURE CODE
    121                                 ;
    122 000000                                  .PSECT
```

```
NTILON     CREATED BY  MACRO  ON 29-JUN-85 AT 03:26       PAGE 4
SYMBOL CROSS REFERENCE                                    CREF   04.00
SYMBOL  VALUE            REFERENCES
.BASEB  = 140000         #5-66
```

```
NTILON     CREATED BY  MACRO  ON 29-JUN-85 AT 03:26       PAGE 5
```

```
LF.BRO= 000400         NF$BLK= 000100         N$$LDV= 000001         P$IPL   000014         S$OWNR  000030
LF.BWT= 000007         NF$DMO= 000010         N$$MCP= 000001         P$LCD   000002         S$PKSZ  000004
LF.ENA= 002000         NF$MOU= 000040         N$$MLL= 000001         P$LEN   000052         S$RCL   000013
LF.LPB= 001000         NF$RST= 000002         N$$MOV= 000010         P$LST   000000         S$STA   000003
LF.MDC= 000100         NF$SCN= 000020         N$$NCT= 000001         P$NRNI  000040         S$TIM   000014
LF.MFL= 004000         NF$SHU= 000004         N$$PEM= 000001         P$OCCB  000050         S$TMR   000016
LF.MTP= 000020         NF$TIM= 000200         PDVTA = ****** GX      P$PFQ   000006         S$WND   000006
LF.PAC= 000200         N$ACQ   000000         PF$BLK= 020000         P$PKSZ  000044         S$$WRG= 000000
LF.RDY= 040000         N$ACTL  000032         PF$CLC= 010000         P$RMX1  000016         S$$YSZ= 007600
LF.REA= 010000         N$ADJ1  000072         PF$DLM= 100000         P$RMX2  000020         S.COST  000001
LF.SER= 000040         N$ADJ2  000074         PF$EIP= 000002         P$RPRI  000042         S.FLG   000000
LF.TIM= 000010         N$CACH  000062         PF$ENB= 000001         P$RTIM  000003         S.LEN   000004
LF.UNL= 020000         N$CIR   000034         PF$FAI= 004000         P$STA1  000022         S.NMST  000002
LF.X2P= 000000         N$CRC   000120         PF$FM1= 000100         P$STA2  000023         S.OWNR  000003
LLCRS = ****** GX      N$DLA   000020         PF$FM2= 000200         P$TIM   000010         TMEFN = 000001
LLCTA = ****** GX      N$DLY   000014         PF$OFF= 000000         P$TSCT  000026         T$FLAG  000044
LN.CLO= 000000         N$ELEN  000054         PF$RM1= 000020         P$TSIZ  000024         T$LIF   000013
LN.DUM= 000005         N$ENC   000042         PF$RM2= 000040         P$TYP   000001         T$LIFL  000013
LN.LOA= 000004         N$ERRC  000022         PF$RVR= 000010         P$$P45= 000000         T$LIFO  000013
LN.LOO= 000003         N$FLG   000005         PF$STA= 000004         P$$WRD= 000000         T$LIFS  000013
LN.OAU= 000003         N$FNC   000006         PF$SVC= 040000         Q$$OPT= 000010         T$LIN   000000
LN.OFF= 000001         N$GENQ  000052         PF$UP = 000004         RF.CTL= 000003         T$LIPS  000006
LN.ON = 000000         N$GTM   000015         PR$BED= 000200         RF.LD1= 000040         T$LLD   000012
LN.OOP= 000004         N$HC1   000052         PR$BEU= 000100         RF.LD2= 000100         T$LLDC  000045
LN.OPE= 000001         N$HC2   000056         PR$BRD= 000040         RF.TIM= 177400         T$LLDL  000012
LN.REF= 000002         N$HIGH  000033         PR$BRU= 000020         RF.TMO= 000400         T$LLDO  000012
LN.SER= 000002         N$LLT   000026         PR$DWN= 000002         RF.WFC= 000200         T$LLDS  000012
LN.STA= 000017         N$LLTM  000024         PR$LCC= 000010         RF.WTD= 000020         T$LLEN  000046
LN.SUB= 000360         N$LVC   000036         PR$MOP= 000004         RF.WTM= 000030         T$LOPR  000002
LN.TRI= 000006         N$LV1   000002         PR$UP = 000001         RF.WTS= 000010         T$LTCL  000024
L$$ASG= 000000         N$LV2   000010         PSIPT = ****** GX      R$$DER= 000000         T$LTIM  000026
L$$DRV= 000000         N$MBXQ  000050         PS$CHR= 000016         R$$K11= 000001         T$LTPR  000014
L$$P11= 000001         N$MHC1  000036         PS$FAI= 000014         R$$SND= 000000         T$LTPS  000020
L$$11R= 000000         N$MHC2  000044         PS$NTI= 000006         R$$11M= 000000         T$NAPL  000004
L.COST  000015         N$PLD   000016         PS$OFF= 000000         SF$BLK= 000001         T$NFE   000000
L.CTL   000012         N$PLLT  000030         PS$STR= 000002         SF$INC= 000002         T$NLEN  000010
L.CVA   177776         N$PRI   000076         PS$UP = 000012         SF.ACT= 000200         T$NNUL  000002
L.DDM   000002         N$ROA1  000022         PS$VER= 000010         SF.ENA= 000100         T$NOPL  000006
L.DDS   000004         N$ROA2  000030         PS$WT = 000004         SF.LPB= 000004         T$NRNI  000042
L.DLC   000003         N$RTMX  000014         PT$BRO= 000200         SF.MFL= 000040         T$NRPL  000005
L.DLM   000006         N$RTM1  000014         PT$DRT= 000100         SF.PAC= 000020         T$NRUL  000007
L.DLS   000010         N$RTM2  000015         PT$END= 000004         SF.REA= 000010         T$NVR   000001
L.FLG   000000         N$RT1   000000         PT$LV1= 000002         SF.SER= 000001         T$RPRI  000040
L.KRBA  000016         N$RT2   000006         PT$LV2= 000001         SF.SVC= 000002         T$SVC   000034
L.LEN = 000022         N$SLA   000016         PT$PH3= 000010         SF.UNL= 000040         T$T5    000030
L.MPF   000022         N$SNOD  000012         PT$XAR= 000020         SLTMA  = ****** GX     T$T6    000032
L.NMST  000020         N$TCTL  000112         PX$BLK= 000040         SLTNM  = ****** GX     T$$KMG= 000000
L.NSTA  000014         N$TIM   000004         PX$DLM= 000200         STAADD  000000R        T$$MIN= 000000
L.OWNR  000021         N$TLC   000100         PX$SVC= 000100         STANUM  000002R        VF$RCV= 100000
L.UNT   000013         N$TNC   000114         P$CHN   000004         STRT   = 000004        VF$XMT= 040000
M$$CRB= 000124         N$TRC   000106         P$CNT   000005         S$ADSZ  000007         V$FLG   000000
M$$CRX= 000000         N$TTCB  000110         P$CTR   000034         S$CTL   000002         V$LEN   000022
M$$EXT= 000000         N$VCB   000010         P$DRCT  000015         S$DTE   000020         V$RCV   000002
M$$FCS= 000000         N$VER   000066         P$DRTR  000036         S$FLG   000010         V$XMT   000012
M$$MGE= 000000         N$XLEN  000124         P$FLG   000012         S$LEN   000032         V$$CTR= 001000
M$$NET= 000000         N$$ACC= 000001         P$FWD   000030         S$LFT   000012         X$$DBT= 000000
M$$OVR= 000000         N$$BUF= 000001         P$ICCB  000046         S$LNK   000000         ZF.COU= 001000
```

```
000110  017700  000000G                  MOV     @SLTNM,R0               ;GET NUMBER OF SYSTEM LINES
000114  017704  000000G                  MOV     @SLTMA,R4               ;GET POINTER TO SLT VECTOR TABLE
000120  012401                   40$:    MOV     (R4)+,R1                ;GET POINTER TO SLT ENTRY
000122  032761  004000  000000           BIT     #LF.MFL,L.FLG(R1)       ;IS LINE MARKED FOR LOAD?
000130  001501                           BEQ     80$                     ;BR IF NO
000132  116102  000002                   MOVB    L.DDM(R1),R2            ;GET THE DDM PDV INDEX
000136  067702  000000G                  ADD     @PDVTA,R2               ;POINT INTO PDV INDEX TABLE
000142  011202                           MOV     (R2),R2                 ;POINT AT PDV FOR DDM PROCESS
000144  032762  002000  000010           BIT     #ZF.PSE,Z.FLG(R2)       ;IS THIS A DLM CIRCUIT?
000152  001404                           BEQ     50$                     ;BR IF NO
000154                                   CALL    DLMMFL                  ;SCAN DLM CIRCUITS FOR MARKED FOR LOAD
000160  103470                           BCS     100$                    ;BR IF ERROR
000162  000464                           BR      80$                     ;
000164  012767  000003  000000G  50$:    MOV     #LS.LIN,$NTLFC+LR.STS   ;SET FUNCTION
000172  105767  000001G                  TSTB    $NIFLG+1                ;TOP-DOWN ALLOCATION?
000176  001403                           BEQ     60$                     ;BR IF NO
000200  052767  002000  000000G          BIS     #LS.TOP,$NTLFC+LR.STS   ;MARK THE REQUEST BLOCK
000206  116167  000012  000004G  60$:    MOVB    L.CTL(R1),$NTLFC+LR.CTL ;GET THE CONTROLLER NUMBER
000214  116167  000013  000005G          MOVB    L.UNT(R1),$NTLFC+LR.UNT ;GET THE UNIT NUMBER
000222  016267  000004  000002G          MOV     Z.NAM(R2),$NTLFC+LR.LIN ; GET THE DDM NAME
000230  016267  000010  000000'          MOV     Z.FLG(R2),PDVFLG        ;SAVE PDV FLAGS
000236                                   CALL    $NINTL                  ;REQUEST NTL TO LOAD PROCESS
000242  103437                           BCS     100$                    ;BR IF NTL NOT INSTALLED
000244  005767  000000G                  TST     $NTLFC                  ;WAS NTL SUCCESSFUL ?
000250  100031                           BPL     80$                     ;IF PL, YES
000252  016767  000002G 000000G          MOV     $NTLFC+LR.LIN,$NIERA    ;SET THE DDM NAME
000260  005067  000002G                  CLR     $NIERA+2                ;MAKE A NULL
000264  116767  000004G 000002G          MOVB    $NTLFC+LR.CTL,$NIERA+2  ;SET CNTL NUMBER
000272  005067  000004G                  CLR     $NIERA+4                ;MAKE A NULL
000276  116767  000005G 000004G          MOVB    $NTLFC+LR.UNT,$NIERA+4  ;SET UNIT NUMBER
000304  032767  000040  000000'          BIT     #ZF.MUX,PDVFLG          ;IS THIS A MULTIPLEXOR?
000312  001003                           BNE     70$                     ;IF NE, YES
000314  052767  100000  000004G          BIS     #100000,$NIERA+4        ;DON'T OUTPUT UNIT #
000322                           70$:    SAVRG   <R0>                    ;SAVE R0
000324                                   EMSG$   30                      ;FAILED TO LOAD LINE ''DEV-CTL-UNT''
000332                                   RESRG   <R0>                    ;RESTORE R0
000334  005300                   80$:    DEC     R0                      ;LOOP TILL DONE
000336  001270                           BNE     40$                     ; ...
000340  000241                   90$:    CLC                             ; SUCCESS!
000342                           100$:   RETURN
```

```
                                .SBTTL  LOCAL DATA
                        ;
                        ; LOCAL DATA
                        ;
000000                          .PSECT  DATA,D

                                .IF DF   R$$MPL
                                .IF NDF  R$$PRO
                        TRNVEC: GIN$     GI.VEC,EXEVEC,VECLN
                        VECLHD: .WORD    0
                                .ENDC
                                .ENDC

000000                  $NIECL::.BLKW                           ;HOLDS PDV INDEX FOR ECL
000002                  $NIXPT::.BLKW                           ;HOLDS PDV INDEX FOR XPT
000004                  $DBECL::.BLKW                           ;HOLDS ADDRESS OF ECL'S DDB
000006                  $DBXPT::.BLKW                           ;HOLDS ADDRESS OF XPT'S DDB

000010                  $NIERA::.BLKW    3                      ;ERROR REPORT ROUTINE ARGUMENT WORDS

000016                  NTIRCX: RCVX$    ,$NIRTN                ;DIRECTIVE TO RCV INTO ''$NIRTN'' OR EXIT
000026                  $NIRTN::.BLKW    2                      ;REQUESTORS TASK NAME (FOLLOW BY $NIRDB)
000032                  $NIRDB::.BLKW    15.                    ;RCV DATA BLOCK (PRECEED BY $NIRTN)

        000001                  MRKTEF = 1                      ;EVENT FLAG FOR MARK TIME
000070                  $NIMRK: MRKT$    MRKTEF,2,2             ;MARK TIME TO WAIT FOR EVL
000102                  $NIEVC::RQST$    EVC...                 ;DIRECTIVE TO REQUEST EVENT COLLECTOR

000120                  $NIAST: .BLKW    1                      ;NOTIFICATION AST ADDRESS
000122                  $NIPRM: .BLKW    1                      ;NOTIFICATION AST PARAMETER TO RETURN

                                .IF NDF  R$$11S
000124                  NTIRCV: RCVD$    ,$NIRTN                ;DIRECTIVE TO RCV DATA INTO ''$NIRTN''

000134                  $NIRNL::RQST$    NTL...                 ;DIRECTIVE TO REQUEST NTL
000152                  $NISNL::SDAT$    NTL...,$NTLFC          ;DIRECTIVE TO SEND ''$NTLFC'' TO NTL
000164                  $NTLFC::.BLKW    3                      ;NTL FUNCTION CODE AND ANSWER

000172                  NTISEN::SDAT$    XXXXXX,$NIFNC          ;DIRECTIVE TO SEND DATA FROM ''$NIFNC''
000204                  $NIFNC::.BLKW    3                      ;FUNCTION REQUEST SAVE WORD
000212                  NTIBSY: SDAT$    XXXXXX,NTIBSC          ;DIRECTIVE TO SEND DATA FROM ''NTIBSC''
000224  000000G         NTIBSC: .WORD    IE.ACT                 ;FAILURE CODE, NTI ALREADY BUSY

000226  000000          $NIFLG::.WORD    0                      ;FLAGS WORD
                                                                ;LOW BYTE:  NON-ZERO = BUSY W/ FUNCTION
                                                                ;HIGH BYTE: FUNCTION-SPECIFIC FLAGS
                                .IFF     ;NDF R$$11S

                                .IF DF   S$$BAS

                        NTISEN: SDAT$    XXXXXX,$NIFNC          ;DIRECTIVE TO SEND DATA FROM ''$NIFNC''
                        $NIFNC::.BLKW    3                      ;FUNCTION REQUEST SAVE WORD

                        ;
                        ; THIS TABLE MAPS THE NTINIT ERROR MESSAGES TO THE DECIMAL NUMBER TO
                        ; BE DISPLAYED ON THE LEDS FOR SERVER BASE SYSTEMS
                        ;
```

```
   613                                          .SBTTL  ERROR MESSAGE PROCESSOR
   614
   615                                  ;+
   616                                  ; $NIERR - FORMAT AND PRINT AN ERROR MESSAGE
   617                                  ;
   618                                  ; THIS ROUTINE DOES NOT NOW SUPPORT SECOND LINE FORMATTING AND/OR PRINTING
   619                                  ;
   620                                  ; INPUTS:
   621                                  ;       $NIERA  = ARGUMENTS FOR ASTERISK REPLACER (IF ONE SPECIFIED)
   622                                  ;       R0      = ADDRESS OF SECOND WORD OF ERROR CONTROL BLOCK
   623                                  ;       BLOCK FORMAT:
   624                                  ;               .WORD   A       ;ADDRESS OF ERROR PROCESSOR ($NIERR)
   625                                  ;               .WORD   0       ;ADDRESS OF FORMAT STRING (NOT USED)
   626                                  ;               .WORD   B OR 0  ;ADDRESS OF ASTERISK REPLACEMENT ROUTINE
   627                                  ;               .WORD   0       ;ADDRESS OF SECOND LINE FORMATTER (NOT USED)
   628                                  ;               .ASCIZ  /MESS/  ;ERROR MESSAGE
   629                                  ;
   630                                  ; OUTPUTS:
   631                                  ;       MESSAGE IS PRINTED, EXCEPT FOR SERVER BASE WHEN EXEC $HALT ROUTINE USED
   632                                  ;       R0 IS DESTROYED
   633                                  ;       CARRY IS SET
   634                                  ;-
   635
   636 000576  010146                   $NIERR::MOV     R1,-(SP)                ;SAVE R1
   637 000600  010246                           MOV     R2,-(SP)                ;SAVE R2
   638 000602  010346                           MOV     R3,-(SP)                ;SAVE R3
   639 000604  010446                           MOV     R4,-(SP)                ;SAVE R4
   640 000606  010546                           MOV     R5,-(SP)                ;SAVE R5
   641
   642                                          .IF DF  S$$BAS
   643
   644                                          TST     -(R0)                   ;POINT TO FIRST WORD IN FORMAT BLOCK
   645                                          MOV     #ERRTAB,R1              ;POINT TO ERROR TABLE
   646                                  10$:    MOV     (R1)+,R2                ;GET NEXT ENTRY IN TABLE (FORMAT ADDRESS)
   647                                          CMP     R2,#TABEND              ;END OF TABLE?
   648                                          BEQ     30$                     ;BR IF END OF TABLE
   649                                          CMP     R2,R0                   ;IS THIS THE CORRECT ENTRY?
   650                                          BEQ     20$                     ;BR IF YES
   651                                          TST     (R1)+                   ;SKIP LED ERROR CODE
   652                                          BR      10$                     ;CONTINUE
   653                                  ;
   654                                  ; FOUND CORRECT ENTRY
   655                                  ;
   656                                  20$:    MOV     (R1)+,R1                ;GET LED ERROR CODE
   657
   658                                          SWSTK$  30$                     ;;ENTER SYSTEM STATE TO HALT PROCESSOR
   659                                          CALL    $HALT                   ;;JUMP TO ROUTINE TO HALT PROCESSOR
   660                                                                          ;;AND DISPLAY ERROR - NEVER RETURNS!!
   661                                  ;
   662                                  ; END OF TABLE
   663                                  ;
   664                                  30$:                                    ;END OF TABLE - IGNORE ERROR
   665
   666                                          .IFF    ;DF S$$BAS
   667
   668 000610  010005                           MOV     R0,R5                   ;COPY ADDRESS IN ERROR BLOCK
   669 000612  012700  000243'                  MOV     #NTIERB,R0              ;GET ADDRESS OF MESSAGE BUFFER
```

```
                                                                 K 15
NTINIT     CREATED BY  MACRO  ON 5-SEP-85 AT 11:04        PAGE 4
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE             REFERENCES

PUTAST     001222 R        9-443   #17-852
QASTT      000632 RG      #7-221    17-875
R$$MPL   = ******          5-65     5-77     5-77     5-77     6-90     7-206    7-228    8-285    10-481
R$$11D   = ******          5-75     5-77
R$$11M   = 000000          5-75
R$$11S   = ******          5-75     6-115    7-189    8-297    9-356    9-364    9-392    9-408    9-422
                           9-438    11-554   15-760   16-794   17-847
R.QSTN   = 000002          6-110    6-118
R.VDTN   = 000002          6-116
R.VXTN   = 000002          6-104
SF.ACT   = 000200         #5-76
SF.ENA   = 000100         #5-76
SF.LPB   = 000004         #5-76
SF.MFL   = 000040         #5-76
SF.PAC   = 000020         #5-76
SF.REA   = 000010         #5-76
SF.SER   = 000001         #5-76
SF.SVC   = 000002         #5-76
SF.UNL   = 000040         #5-76
SLTMA      000646 RG      #7-232
SLTNM      000650 RG      #7-233
SRSTD      000616 RG      #7-215    17-855
SWR      = 177570         #5-77
S$$BAS   = ******          7-190    7-190    7-192    7-192    7-193    7-193    7-194    7-194    8-312
                           9-373    9-407    13-642   14-711   17-848
S.COST     000001         #5-76
S.DATN   = 000002          6-119    6-122    6-124   *16-811  *16-812  *16-820  *16-821   17-854
S.FLG      000000         #5-76
S.LEN      000004         #5-76
S.NMST     000002         #5-76
S.OWNR     000003         #5-76
TKPS       000636 RG      #7-223
TKTCB      000630 RG      #7-220
TK100      000662 RG      #7-238
TPS      = 177564         #5-77
T1SCL      000674 RG      #7-243
T100C      000656 RG      #7-236
T100Q      000660 RG      #7-237
UBMPR    = 170200         #5-77
UISAR0   = 177640         #5-77
UISAR1   = 177642         #5-77
UMRPT      000614 RG      #7-214
VECLN    = 000043  G      #7-252
X$$MCB   = ******          5-75     5-75
ZF.COU   = 001000         #5-75
ZF.DDM   = 000001         #5-75
ZF.DIA   = 004000         #5-75
ZF.DLC   = 000002         #5-75
ZF.DVP   = 100000         #5-75
ZF.INI   = 040000         #5-75
ZF.KMX   = 000020         #5-75
ZF.LLC   = 000004         #5-75
```

```
                                                                 L 15
NTINIT     CREATED BY  MACRO  ON 5-SEP-85 AT 11:04        PAGE 5
```

```
NTITIM  MACRO V05.03b Saturday 29-Jun-85 03:28  Page 9-1
Symbol table

ABTIM = ****** GX       C.SRC   000014          K$$CNT= 177546          N$$MOV= 000010          S$$WRG= 000000
A$$CHK= 000000          C.SSHT= 000004          K$$CSR= 177546          N$$NCT= 000001          S$$YSZ= 007600
A$$CPS= 000000          C.SUB   000012          K$$LDC= 000000          N$$PEM= 000001          TIFLG   000000R
A$$PRI= 000000          C.SYST= 000006          K$$TPS= 000074          PIRQ  = 177772          TKPS  = ****** GX
A$$TRP= 000000          C.SYTK= 000010          LD$LP = 000000          PMODE = 030000          TK100 = ****** GX
BUFUMP= 172354          C.TCB   000004          L$$ASG= 000000          PR0   = 000000          TPS   = 177564
CLINS   000162R         C.TIM   000006          L$$DRV= 000000          PR1   = 000040          T$$KMG= 000000
CLKHD = ****** GX       C.UIC   000016          L$$P11= 000001          PR2   = 000100          T$$MIN= 000000
CMODE = 140000          D$$BUG= 177514          L$$11R= 000000          PR3   = 000140          T1    = 000002
C$$CKP= 000000          D$$ISK= 000000          MPAR  = 172100          PR4   = 000200          T1SCL = ****** GX
C$$ORE= 000400          D$$L11= 000001          MPCSR = 177746          PR5   = 000240          T1SIN   000116R
C$$RSH= 177564          D$$YNC= 000000          M$$CRB= 000124          PR6   = 000300          T100C = ****** GX
C.AR5   000014          D$$YNM= 000000          M$$CRX= 000000          PR7   = 000340          T100Q = ****** GX
C.AST   000012          E$$XPR= 000000          M$$FCS= 000000          PS    = 177776          T5    = 000001
C.CSTP= 000012          F$$LVL= 000001          M$$MGE= 000000          P$$P45= 000000          T50IN   000130R
C.DST   000016          G$$TPP= 000000          M$$NET= 000000          P$$WRD= 000000          UBMPR = 170200
C.EFN   000003          G$$TSS= 000000          M$$OVR= 000000          Q$$OPT= 000010          UISAR0= 177640
C.LGTH= 000020          G$$TTK= 000000          N$$ACC= 000001          R$$DER= 000000          UISAR1= 177642
C.LNK   000000          G$$WRD= 000000          N$$BUF= 000001          R$$K11= 000001          V$$CTR= 001000
C.MRKT= 000000          I$$RAR= 000000          N$$LDV= 000001          R$$SND= 000000          X$$DBT= 000000
C.RQT   000002          I$$RDN= 000000          N$$MCP= 000001          R$$11M= 000000          $NITIM  000002RG
C.RSI   000012          KISAR0= 172340          N$$MLL= 000001          SWR   = 177570          .BASEB= 140000
C.SCHD= 000002          KISAR6= 172354

. ABS.  000020    000   (RW,I,GBL,ABS,OVR)
        000250    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10026 Words  ( 40 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:09.06
SY:NTITIM.V2,[132,134]NTITIM/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTITIM
```

```
NTITIM     CREATED BY  MACRO  ON 29-JUN-85 AT 03:28     PAGE 1
```

```
PF$FM2= 000200          P$FLG   000012          R.NAM   000002          T$NNUL  000002          T3.NSD= 000200
PF$OFF= 000000          P$FWD   000030          SCAN    002266R         T$NOPL  000006          T3.PMD= 040000
PF$RM1= 000020          P$ICCB  000046          SPARE = 000010          T$NRNI  000042          T3.PRV= 010000
PF$RM2= 000040          P$IPL   000014          SP.EIP= 000001          T$NRPL  000005          T3.REM= 020000
PF$RVR= 000010          P$LCD   000002          SP.ENB= 000002          T$NRUL  000007          T3.ROV= 000040
PF$STA= 000004          P$LEN   000052          SP.LOG= 000004          T$NVR   000001          T3.RST= 000400
PF$SVC= 040000          P$LST   000000          SYSFDB= ****** GX       T$RPRI  000040          T3.SLV= 002000
PF$UP = 000004          P$NRNI  000040          S$$WRG= 000000          T$SVC   000034          T3.SWS= 000002
PRLTA   000200R    002  P$OCCB  000050          S$$YSZ= 007600          T$T5    000030          UBA     000010R    002
PR$BED= 000200          P$PFQ   000006          S.BMSK  177776          T$T6    000032          UBIAS   000006R    002
PR$BEU= 000100          P$PKSZ  000044          S.BMSV  177774          T$$KMG= 000000          UNITS   000004R    002
PR$BRD= 000040          P$RMX1  000016          S.CON   000010          T$$MIN= 000000          UPTR    000112R    002
PR$BRU= 000020          P$RMX2  000020          S.CSR   000012          T.ACTL  000052          VECT    002402R
PR$DWN= 000002          P$RPRI  000042          S.CTM   000006          T.ASTL  000016          VF$RCV= 100000
PR$LCC= 000010          P$RTIM  000003          S.DMCS  000020          T.ATT   000062          VF$XMT= 040000
PR$MOP= 000004          P$STA1  000022          S.FRK   000016          T.CPCB  000004          VF.LIB= 000001
PR$UP = 000001          P$STA2  000023          S.ITM   000007          T.DPRI  000040          VF.SS = ****** GX
PR.LTA= 000001          P$TIM   000010          S.LHD   000000          T.EFLG  000022          V$FLG   000000
PR.SEC= 000002          P$TSCT  000026          S.PKT   000014          T.IOC   000003          V$LEN   000022
PR7   = ****** GX       P$TSIZ  000024          S.PRI   000004          T.LBN   000041          V$RCV   000002
PS$CHR= 000016          P$TYP   000001          S.RCNT  177772          T.LDV   000044          V$XMT   000012
PS$ AI= 000014          P$$P45= 000000          S.ROFF  177773          T.LNK   000000          V$$CTR= 001000
PS$NTI= 000006          P$$WRD= 000000          S.STS   000011          T.MXSZ  000050          V.ADD   000006
PS$OFF= 000000          P.BLKS  000016          S.VCT   000005          T.NAM   000006          V.BIAS  000004
PS$STR= 000002          P.BUSY  000024          TSKSCH  006260R         T.OFF   000066          V.FLG   000002
PS$UP = 000012          P.IOC   000003          TS.BLK= 171700          T.PCB   000046          V.LEN   000012
PS$VER= 000010          P.LNK   000000          TS.CKP= 000200          T.PRI   000002          V.LNK   000000
PS$WT = 000004          P.MAIN  000012          TS.CKR= 000100          T.RCVL  000012          V.PDV   000004
PS.APR= 000007          P.NAM   000004          TS.EXE= 100000          T.RRFL  000072          V.SIZ   000010
PS.CHK= 010000          P.OWN   000026          TS.HLD= 002000          T.SAST  000054          XALITB  004660R
PS.CKP= 040000          P.PRI   000002          TS.MSG= 020000          T.SRCT  000071          XDCB    004456R
PS.CKR= 020000          P.REL   000014          TS.NRP= 010000          T.STAT  000032          XDLC    000360RG
PS.COM= 000200          P.SIZE  000016          TS.OUT= 000400          T.ST2   000034          XFLTTB  005150R
PS.DEL= 000010          P.STAT  000030          TS.RDN= 040000          T.ST3   000036          XKRB    002162R
PS.DRV= 000020          P.SUB   000010          TS.RUN= 004000          T.TCBL  000030          XLIN    001614R
PS.FXD= 004000          P.SWSZ  000022          TS.STP= 001000          T.TIO   000057          XLT     005456R
PS.LIO= 001000          P.TCB   000026          T$FLAG  000044          T.TKSZ  000060          XLT16   005452R
PS.NSF= 000400          P.WAIT  000020          T$LIF   000013          T.UCB   000026          XOBJTB  004576R
PS.OUT= 100000          Q$$OPT= 000010          T$LIFL  000013          T2.ABO= 000100          XPRO    000152P
PS.PER= 002000          RF.CTL= 000003          T$LIFO  000013          T2.AST= 100000          XREMTB  005022R
PS.PIC= 000100          RF.LD1= 000040          T$LIFS  000013          T2.CAF= 000400          X$$DBT= 000000
PS.SYS= 000040          RF.LD2= 000100          T$LIN   000000          T2.CHK= 020000          ZDDS    003354R
PT$BRO= 000200          RF.LOO= 100000          T$LIPS  000006          T2.CKD= 010000          ZF.LLC= ****** GX
PT$DRT= 000100          RF.TIM= 177400          T$LLD   000012          T2.DST= 040000          ZF.LMC= ****** GX
PT$END= 000004          RF.TMO= 000400          T$LLDC  000045          T2.FXD= 002000          ZF.MFL= ****** GX
PT$LV1= 000002          RF.WFC= 000200          T$LLDL  000012          T2.HLT= 000200          ZF.PSE= ****** GX
PT$LV2= 000001          RF.WTD= 000020          T$LLDO  000012          T2.REX= 001000          Z.AVL = ****** GX
PT$PH3= 000010          RF.WTM= 000030          T$LLDS  000012          T2.SEF= 004000          Z.DAT = ****** GX
PT$XAR= 000020          RF.WTS= 000010          T$LLEN  000046          T2.SPN= 000004          Z.DSP = ****** GX
PX$BLK= 000040          R$$DER= 000000          T$LOPR  000002          T2.STP= 000020          Z.FLG = ****** GX
PX$DLM= 000200          R$$K11= 000001          T$LTCL  000024          T2.WFR= 000001          Z.NAM = ****** GX
PX$SVC= 000100          R$$SND= 000000          T$LTIM  000026          T3.ACP= 100000          Z.PCB = ****** GX
P$CHN   000004          R$$11M= 000000          T$LTPR  000014          T3.CAL= 000100          $ADRX = ****** GX
P$CNT   000005          R.ADD   000010          T$LTPS  000020          T3.CLI= 001000          $ADRZ = ****** GX
P$CTR   000034          R.FLAG  000012          T$NAPL  000004          T3.GFL= 000010          $BFR  = ****** GX
P$DRCT  000015          R.LEN   000014          T$NFE   000000          T3.MCR= 004000          $BIAS = ****** GX
P$DRTA  000036          R.LNK   000000          T$NLEN  000010          T3.NET= 000020          $CCBLH= ****** GX
```

```
                                        .TITLE  VUNPSI - VNP UNLOAD ROUTINES FOR PSI
                                        .IDENT  /V05.00/
                                        .ENABL  LC                                                       ;[TD001]
                                                                                                         ;**-1
                                ;
                                ; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ;       VNP - PSI UNLOAD ROUTINES
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  05-MAR-82
                                ;       DECNET-11M/S V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 3.00  16-APR-82
                                ;       DECNET-11M V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;

```

```
                                        .SBTTL  XTRA - DEALLOCATE TRACE DATA STRUCTURES

                                ;+
                                ; XTRA - DEALLOCATE TRACE STRUCTURES
                                ;
                                ; INPUTS
                                ;       NONE
                                ;
                                ; OUTPUTS
                                ;       TRACE STRUCTURES ARE DEALLOCATED
                                ;
                                ;
                                ;-

001554                          XTRA:
001554                                  CALL    $SAVAL            ; SAVE ALL REGISTERS

001560  016703  000000G                 MOV     $PSIPT,R3         ; GET HOME BLOCK ADDRESS
001564  062703  000016                  ADD     #H$TRB,R3         ; POINT TO TRACE CONTROL BLOCK ADDRESS
001570                                  GETRV   R3,#2             ; READ IN TRACE CONTROL BLOCK ADDRESS
001606  016700  000000G                 MOV     $BFR,R0           ; GET TRACE CONTROL BLOCK ADDRESS
001612  001441                          BEQ     30$               ; BR IF NOT TRACING
001614  005067  000000G                 CLR     $BFR              ; CAUSE TRACE TASK TO EXIT
001620                                  PUTRC                     ; ...
001624                                  GETRV   R0,#T$CLEN        ; READ IN TRACE CONTROL BLOCK

001642                          10$:    SAVRG   <R0>              ; SAVE ADDRESS OF BLOCK
001644                                  SAVRG   <#T$CLEN>         ; SAVE LENGYH OF SAME

001650  016701  000010G                 MOV     T$CSIZ+$BFR,R1    ; GET SIZE OF BUFFER IN BLOCKS
001654                                  ASL$    6,R1              ; COMPUTE SIZE IN BYTES
001670  010002                          MOV     R0,R2             ; COPY END POINTER
001672  061602                          ADD     (SP),R2           ; COMPUTE END ADDRESS OF TRACE BLOCK
001674  026702  000004G                 CMP     T$CBUF+$BFR,R2    ; IS BUFFER CONTIGUOUS WITH CONTROL BLOCK?
001700  001401                          BEQ     20$               ; YES, DEALLOCATE AS ONE

                                        ;*** DEALLOCATE NON-POOL TRACE BUFFER HERE (HOW?) ***
001702  000401                          BR      25$               ; MERGE TO DEALLOCATE CONTROL BLOCK

001704  060116                  20$:    ADD     R1,(SP)           ; INCLUDE LENGTH OF BUFFER IN DEALLOC
001706                          25$:    RESRG   <R1>              ; GET LENGTH TO DEALLOCATE
001710                                  RESRG   <R0>              ; GET CONTROL BLOCK ADDRESS
001712                                  CALL    $DEA16            ; DEALLOCATE TO POOL
001716                          30$:    RETURN                    ; RETURN TO USER MODE AND CALLER
```

```
                                        ;+
                                        ; $TMLTA - SETUP DATA BASE ADDRESS IN SYSTEM LINE TABLE
                                        ;
                                        ; INPUTS:
                                        ;       $SADDR=DATA BASE ADDRESS
                                        ;       $MXADD=MUX TABLES ADDRESS
                                        ;       $OFF=OFFSET VALUE
                                        ;       $SLTA=SYSTEM LINE TABLE ADDRESS
                                        ;       F1.ERR SET IF ANY ERRORS OCCURRED
                                        ;
                                        ; OUTPUTS:
                                        ;       R0,R1=DESTROYED
                                        ;       L.DDS AND/OR L.DLS SET TO DATA BASE ADDRESS
                                        ;       LF.RDY SET IF APPROPRIATE
                                        ;-
000142                                  $TMLTA::
000142  005767  000000G                         TST     $FLAG1          ; HAVE ANY ERRORS OCCURRED ?
000146  100531                                  BMI     80$             ; YES .. DO NOTHING
000150  016700  000000G                         MOV     $SLTA,R0        ; GET SYSTEM LINE TABLE ADDRESS
000154  032767  000000G 000000G                 BIT     #FL.CHA,$FLAGS  ; MUX CHARACTERISTICS UPDATE ONLY ?
000162  001121                                  BNE     70$             ; YES .. MARK THE LINE READY

000164  032767  000000G 000000G                 BIT     #FL.DLC,$FLAGS  ; IS THIS PROCESS A DLC ?
000172  001417                                  BEQ     5$              ; IF EQ, NO .. NORMAL DATA BASE LOAD
000174  022767  120000  000000G                 CMP     #120000,$SADDR  ; IS THE LINE TABLE MAPPED ?
000202  101013                                  BHI     5$              ; IF HI, NO .. NORMAL DATA BASE
000204  016701  000000G                         MOV     $DLCPV,R1       ; ASSUME IT'S NOT A DDM
000210  032767  000000G 000000G                 BIT     #FL.DDM,$FLAGS  ; IS THIS A DDM ?
000216  001402                                  BEQ     3$              ; IF EQ, NO
000220  016701  000000G                         MOV     $PDVA,R1        ; GET THE PDV ADDRESS FOR THIS PROCESS
000224  016160  000000G 000000G 3$:             MOV     Z.DSP(R1),L.DLM(R0) ; SET THE DATA BASE BIAS

000232  016701  000000G         5$:             MOV     $SADDR,R1       ; GET DATA BASE ADDRESS
000236  001475                                  BEQ     80$             ; IF EQ, DO NOTHING
000240  066701  000000G                         ADD     $OFF,R1         ; ADD OFFSET VALUE
000244  032767  000000G 000000G                 BIT     #FL.DDM,$FLAGS  ; ARE WE LOADING A DDM?
000252  001011                                  BNE     10$             ; IF NE, YES
000254  032767  000000G 000000G                 BIT     #FL.DLC,$FLAGS  ; ARE WE LOADING A DLC?
000262  001057                                  BNE     60$             ; IF NE, YES
000264  016700  000000G                         MOV     $PDVA,R0        ; GET PDV ADDRESS
000270  010160  000000G                         MOV     R1,Z.DAT(R0)    ; SET LLC DATA BASE ADDRESS
000274  000456                                  BR      80$
000276  032767  000000G 000000G 10$:            BIT     #FL.MUX,$FLAGS  ; IS THIS A MUX LINE?
000304  001440                                  BEQ     50$             ; IF EQ, NO
000306  005767  000000G                         TST     $MXADD          ; MUX TABLES SEPARATE ?
000312  001402                                  BEQ     15$             ; IF EQ, NO
000314  016701  000000G                         MOV     $MXADD,R1       ; ELSE, POINT TO MUX TABLES

000320  016746  000000G         15$:            MOV     $SLTNM,-(SP)    ; GET NUMBER OF SLT ENTRIES ON STACK
000324  016703  000000G                         MOV     $SLTMA,R3       ; GET SLT TABLE ADDRESS
000330  012302                  20$:            MOV     (R3)+,R2        ; GET NEXT SLT ADDRESS
000332  126062  000000G 000000G                 CMPB    L.DDM(R0),L.DDM(R2) ; DO DDM PDV INDICES MATCH?
000340  001016                                  BNE     40$             ; IF NE, NO
000342  126062  000000G 000000G                 CMPB    L.CTL(R0),L.CTL(R2) ; DO CONTROLLER NUMBERS MATCH?
000350  001012                                  BNE     40$             ; IF NE, NO
000352  010162  000000G                         MOV     R1,L.DDS(R2)    ; SET DDM LINE TABLE ADDRESS
000356  126262  000000G 000000G                 CMPB    L.DDM(R2),L.DLC(R2) ; COMBINATION DDM/DLC?
```

```
                                                                 L 5
VXIT - VNP EXIT ROUTINES        MACRO V05.03b Saturday 29-Jun-85 03:12  Page 5-3
Symbol table

A$$CHK= 000000          D$MAXV  000070          F.NRBD= ****** GX       N$$BUF= 000001          $DEA16= ****** GX
A$$CPS= 000000          D$MLL   000040          F.RSIZ= ****** GX       N$$LDV= 000001          $DECPT= ****** GX
A$$PRI= 000000          D$MNOD  000041          F.URBD= ****** GX       N$$MCP= 000001          $DEVHD= ****** GX
A$$TRP= 000000          D$NA    000062          G$$TPP= 000000          N$$MLL= 000001          $EROUT= ****** GX
C$$CKP= 000000          D$NBEA  000056          G$$TSS= 000000          N$$MOV= 000010          $FMASK= ****** GX
C$$ORE= 000400          D$NBRA  000054          G$$TTK= 000000          N$$NCT= 000001          $GETRC= ****** GX
C$$RSH= 177564          D$NEND= 000054          G$$WRD= 000000          N$$PEM= 000001          $GETRV= ****** GX
D$AMXC  000072          D$NLN   000030          I$$RAR= 000000          N.FID = ****** GX       $PARHD= ****** GX
D$AMXH  000074          D$NN    000060          I$$RDN= 000000          PAR$$$= 000027          $PUTRC= ****** GX
D$ANN   000000          D$OUTT  000043          K$$CNT= 177546          P$$P45= 000000          $PUTRD= ****** GX
D$BRPR  000102          D$RETF  000050          K$$CSR= 177546          P$$WRD= 000000          $RADDR= ****** GX
D$BRTM  000100          D$RNN   000002          K$$LDC= 000000          Q$$OPT= 000010          $RBLCK= ****** GX
D$DELF  000045          D$RTMR  000076          K$$TPS= 000074          R$$DER= 000000          $RSIZ = ****** GX
D$DELW  000046          D$SEG   000036          LD$LP = 000000          R$$K11= 000001          $RSIZE= ****** GX
D$END = 000104          D$SER   000032          L$BBLK= ****** GX       R$$SND= 000000          $VFLAG= ****** GX
D$FNB   000034          D$SQRL  000052          L$BXFR= ****** GX       R$$11M= 000000          $VFLG1= ****** GX
D$HIOR  000024          D$$BUG= 177514          L$$ASG= 000000          SYSFDB= ****** GX       $VIMO = ****** GX
D$HOST  000022          D$$ISK= 000000          L$$DRV= 000000          S$$WRG= 000000          $VXIT   000000RG
D$INAC  000044          D$$L11= 000001          L$$P11= 000001          S$$YSZ= 007600          $XFRAD= ****** GX
D$INCT  000042          D$$YNC= 000000          L$$11R= 000000          T$$KMG= 000000          $$    = 177777
D$IPL   000051          D$$YNM= 000000          M$$CRB= 000124          T$$MIN= 000000          $$$   = 000354R
D$LID   000020          ERR1    000426R         M$$CRX= 000000          VF.INI= ****** GX       $$$$  = 000426R
D$LNAM  000006          E$$XPR= 000000          M$$FCS= 000000          V$$CTR= 001000          .CLOSE= ****** G
D$LNUM  000014          FE.CEX= ****** GX       M$$MGE= 000000          V1.NSY= ****** GX       .CRAVL= ****** GX
D$LST   000047          F$$LVL= 000001          M$$NET= 000000          X$$DBT= 000000          .DEVHD= ****** GX
D$MAXC  000064          F.ERR = ****** GX       M$$OVR= 000000          $BFR  = ****** GX       .FMASK= ****** GX
D$MAXH  000066          F.FNB = ****** GX       N$$ACC= 000001          $CRAVL= ****** GX       .PARHD= ****** GX


. ABS.  000104    000   (RW,I,GBL,ABS,OVR)
        000432    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 11384 Words  ( 45 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:11.43
SY:VXIT.V2,[132,134]VXIT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VXIT
```

```
                                                   M 5
VXIT       CREATED BY  MACRO  ON 29-JUN-85 AT 03:12     PAGE 1
```

```
        .TITLE  VXPR - VNP RELEASE PROCESS FROM PARTITION ROUTINES
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       VNP - RELEASE PROCESS FROM PARTITION ROUTINE
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                                          MOV     #$NIDLI,R0              ;GET DOWN-LINE LOAD INITIALIZER
                                          BEQ     5$                      ;BR IF NO SUCH MODULE
                                          CALL    (R0)                    ;PERFORM DOWN-LINE LOAD INIT
                                  5$:     .ENDC                           ;*-END RSX11S CODE-*

                                          .IF DF  R$$MPL                                                  ;[MP01]
                                          CALL    INILIN                  ; REINIT HRC... IF RSX DATA STRUCTURES
                                                                          ; HAVE CHANGED                  ;[MP01]
                                          .ENDC                                                           ;[MP01]

000216                                    SWSTK$  10$                     ;ENTER KERNEL MODE
000222  017777  000000G  000000G          MOV     @NTLPT,@KSAR6           ;;MAP TO THE NETLDR DATA BLOCK

                                          .IF      DF,M$$EXT
000230  016766  140002'  000006           MOV     .BASEB+2,6(SP)          ;;RETURN UMR BLOCK ADDRESS IN R2
                                          .ENDC

000236  016766  140000G  000010           MOV     .BASEB+.CXCSR,10(SP)    ;;RETURN CSR CHECK BLOCK ADDRESS IN R3
000244                                    RETURN                          ;;BACK TO USER MODE AT 10$
000246                            10$:
                                          .IF      DF,M$$EXT

000246  005702                            TST     R2                      ;IS THERE A UMR BLOCK ?
000250  001406                            BEQ     20$                     ;IF EQ, NO

                                            .IF DF R$$MPL
                                          BIT     #HF.UBM,@HFMSK          ;DOES THIS SYSTEM HAVE UMR'S?
                                          BEQ     20$                     ;BR IF NO
                                            .IFF         ; DF R$$MPL
000252  005777  000000G                   TST     @UMRPT                  ;DOES THIS SYSTEM HAVE UMR'S?
000256  001403                            BEQ     20$                     ;BR IF NO
                                            .ENDC        ; DF R$$MPL

000260                                    CALL    $NIUMR                  ;LOAD THE UMR'S
000264  000402                            BR      25$                     ; CONTINUE
                                          .ENDC

000266                            20$:    .IF DF  R$$MPL

                                          SWSTK$  25$                     ;;ENTER SYSTEM STATE IF M+
                                                                          ;;(READ/WRITE ACCESS TO LOW MEMORY)
                                          .IFTF   ;DF R$$MPL

000266  005077  000000G                   CLR     @PUMR                   ;INDICATE NO UMR'S

                                          .IFT    ;DF R$$MPL

                                          RETURN                          ;;RETURN TO USER STATE IF M+

                                          .ENDC

000272  012700  000000G           25$:    MOV     #$NICSR,R0              ;GET CSR CHECK ROUTINE ADDRESS
000276  001401                            BEQ     30$                     ;BR IF NO SUCH ROUTINE
000300                                    CALL    (R0)                    ;CHECK CSR VALIDITY

000302                            30$:
                                          .IF DF  R$$MPL                  ; SPECIAL CASE HACK FOR NX      ;[MP01]
                                          .SBTTL  CSR CHECK

                                  ;+
                                  ; $NICSR - CHECK IF ALL CSR'S ARE ON-LINE
                                  ;
                                  ; CSR CHECK LIST FORMAT:
                                  ;
                                  ;       .WORD   LINK            ;LINK TO NEXT BLOCK IN LIST OR ZERO (EOL)
                                  ;       .REPT   15.             ;SERIES OF 2 WORD ENTRIES, 0 <= X <= 15.
                                  ;        .WORD  SLTA            ; SYSTEM LINE TABLE ADDRESS
                                  ;        .WORD  CSRA            ; CSR ADDRESS
                                  ;       .ENDR                   ;NUMBER OF ENTRIES MAY RANGE FROM 0. TO 15.
                                  ;
                                  ; INPUTS:
                                  ;       R3 = ADDRESS (BIAS) OF FIRST BLOCK IN CSR CHECK LIST (ZERO IF NONE)
                                  ;
                                  ; OUTPUTS:
                                  ;       ALL CSR'S IN LIST ARE CHECKED.
                                  ;       R3 IS DESTROYED.
                                  ;-

000000  005703                    $NICSR::TST     R3                      ;CLEAR CARRY, ANY CSR'S TO CHECK?
000002  001533                            BEQ     60$                     ;BR IF NO, FAST EXIT
000004                                    SAVRG   <R0,R1,R2>              ;SAVE R0,R1,R2
000012  012702  000017                    MOV     #17,R2                  ;SET INITIAL MAX COUNT
000016  012701  140002                    MOV     #.BASEB+2,R1            ;R1 POINTS AT ENTRY TO CHECK
000022  005000                    10$:    CLR     R0                      ;R0 WILL RETURN CSR ADDRESS OR ZERO
000024                                    SWSTK$  30$                     ;GO TO SYSTEM STATE
000030  010377  000000G                   MOV     R3,@KSAR6               ;MAP TO CURRENT BLOCK IN LIST
000034  005302                    15$:    DEC     R2                      ;COUNT DOWN NUMBER OF ENTRIES
000036  002014                            BGE     20$                     ;BR IF ANOTHER TO CHECK
000040  016766  140000' 000010            MOV     .BASEB,10(SP)           ;RESET USER R3 TO NEXT BLOCK
000046  001423                            BEQ     25$                     ;BR IF NO NEXT BLOCK, END OF LIST
000050  016777  140000' 000000G           MOV     .BASEB,@KSAR6           ;MAP TO NEXT BLOCK
000056  012701  140002                    MOV     #.BASEB+2,R1            ;POINT AT FIRST ENTRY
000062  012702  000017                    MOV     #17,R2                  ;RESET MAX COUNT
000066  000762                            BR      15$                     ;GO CHECK IT

000070  012167  000016'           20$:    MOV     (R1)+,CSRSLT            ;PULL OUT THE SLT ADDRESS
000074  001002                            BNE     23$                     ;BR IF ENTRY IS NOT EMPTY
000076  005721                            TST     (R1)+                   ;SKIP SECOND WORD
000100  000755                            BR      15$                     ;GET NEXT ENTRY

000102  012166  000002            23$:    MOV     (R1)+,2(SP)             ;SET USER R0 TO CSR ADDRESS
000106  010166  000004                    MOV     R1,4(SP)                ;POINT USER R1 AT NEXT ENTRY
000112  010266  000006                    MOV     R2,6(SP)                ;PUT CURRENT COUNT IN USER R2

000116                            25$:    RETURN                          ;RETURN TO USER MODE AT "30$"

000120  005700                    30$:    TST     R0                      ;GOT CSR ADDRESS OR END OF LIST?
000122  001457                            BEQ     50$                     ;BR IF FOUND END OF LIST

000124  005046                            CLR     -(SP)                   ;USE AS ARGUMENT FOR TRAP ROUTINE
000126                                    DIR$    #CSRENA                 ;ENABLE ILLEGAL ADDRESS TRAPPING
000134  005710                            TST     (R0)                    ;CHECK THE CSR ADDRESS
000136                                    DIR$    #CSRDIS                 ;DISABLE TRAPPING
000144  006026                            ROR     (SP)+                   ;WAS CSR ADDRESS ILLEGAL?
```

```
                                          .SBTTL  SET EXECUTOR ON

                              ;+
                              ; $NILON - SET STATE EXECUTOR ON
                              ;
                              ; REQUEST THE NETACP TO START
                              ;
                              ; INPUTS:
                              ;         NETACP INSTALLED
                              ;
                              ; OUTPUTS:
                              ;         THE DEVICE LIST IS SCANNED TO FIND THE NETACP.  IF IT IS FOUND,
                              ;         A VCB IS ALLOCATED FOR IT, AND IT IS REQUESTED TO RUN.
                              ;         $NIRDB IS SET TO INDICATE SUCCESS (+) OR FAILURE (-) OR NO-OP (0).
                              ;         CARRY IS SET ON FAILURE.
                              ;-
000000                        $NILON::
000000                                  CALL    FNDPDV                  ;FIND NSP PROCESSES (ECL & XPT)
000004  005767  000000G                 TST     $DBXPT                  ;BOTH TRANSPORT (XPT)
000010  001403                          BEQ     10$                     ;  AND ECL MUST BE
000012  005767  000000G                 TST     $DBECL                  ;    PRESENT IN THE SYSTEM
000016  001002                          BNE     20$                     ;      AND BE LOADED

000020  000261                10$:      SEC
000022                                  RETURN

000024                        20$:      DIR$    #ASSIGN                 ;ASSIGN LUN 2 TO NS:
000032  103003                          BCC     30$                     ; IF CS, ERROR
000034                                  EMSG$R  15                      ;ALUN$ FAILURE OR NS: LOAD FAILURE

000042  016700  000000G       30$:      MOV     $NIXPT,R0               ;GET PDV INDEX OF XPT
000046  067700  000000G                 ADD     @PDVTA,R0               ;POINT INTO PDV INDEX TABLE
000052  011000                          MOV     (R0),R0                 ;GET ADDRESS OF XPT'S PDV
000054  017701  000000G                 MOV     @DECPT,R1               ;GET DECNET HOME BLOCK ADDRESS
000060  116061  000006  0000            MOVB    Z.LLN(R0),D$NLN(R1)     ;INITIALIZE THE NUMBER OF ROUTIN

000066                                  SWSTK$  60$                     ;* ENTER SYSTEM STATE
000072  005066  000004                  CLR     4(SP)                   ;* CLEAR USER R1

                                        .IF     NDF,R$$MPL
000076  017703  000000G                 MOV     @HEADR,R3               ;* GET TASK HEADER
000102  016303  000102                  MOV     H.LUN+4(R3),R3          ;* ADDRESS OF NS: UCB
                                        .IFF
                                        SAVMAP                          ;* SAVE APR6 CONTENTS
                                        MAP     @SAHDB                  ;* MAP TO HEADER IN SECONDARY PO
                                        MOV     @SAHPT,R3               ;* GET THE HEADER POINTER
                                        MOV     H.LUN+4(R3),R3          ;* ADDRESS OF NS: UCB
                                        RESMAP                          ;* RESTORE APR6
                                        .ENDC

000106  016301  000014                  MOV     U.CW3(R3),R1            ;* GET FEATURES WORD FOR THIS NODE
000112  032701  000200                  BIT     #NF.END,R1              ;* IS THIS A FULL ROUTING NODE?
000116  001020                          BNE     50$                     ;* IF NE, NO
000120  012703  000004'                 MOV     #RCP1NM,R3              ;* POINT TO TASK NAME
000124                                  CALL    @SRSTD                  ;* IS ROUTING CONTROL PROGRAM IN
000130  103407                          BCS     40$                     ;* IF CS, NOT INSTALLED

```

```
                                                                  L 10
NTILON    CREATED BY  MACRO  ON 29-JUN-85 AT 03:26       PAGE 5
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME        REFERENCES

ADJDF$           #5-62      5-73
ALUN$            #5-61      6-105
BIAS             #5-61
CACDF$           #5-62      5-74
CALL              8-141     8-160     8-178     8-185     8-206     8-212     8-213     8-253     8-254     8-276
                  8-285     9-315     9-317
CALLR             8-288
CALLX            #5-62
CCBDF$           #5-63      5-65
CTRDF$           #5-61      5-71
DCBDF$           #5-63      5-77
DHBDF$           #5-62      5-75
DIR$             #5-60      8-150     8-265    #8-274     8-274
DVBDF$           #5-63      5-76
ECDDB$           #5-61      5-69
EMSG$            #5-60
EMSG$R           #5-60      8-152     8-192     8-195     8-231     8-294     8-295     8-296     8-297
HDRDF$           #5-63      5-79
MAP              #5-61
NHWDF$           #5-60      5-66
NSFDF$           #5-62      5-72
NTLER$           #5-60      7-112     7-113     7-114     7-115     7-116
OFF$             #6-104    #6-105
PDVDF$           #5-61      5-70
PLBDF$           #5-61      5-67
RESMAP           #5-62
RETC             #5-60      9-320
RETURN            8-148     8-188     8-227     8-260     8-278     9-325
RUN$             #5-60      6-104
R50$             #6-104     6-104
SAVMAP           #5-62
SOB               8-241     8-259
SWSTK$            8-160     8-212     8-253     8-285     9-315
UCBDF$           #5-63      5-78
WSIG$S           #5-60      8-274
XPDDB$           #5-61      5-68
```

```
                                                                  M 10
**FILE**ID**NTIMFE
```

```
                                                                                   L 11
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:08  Page 8-4
Symbol table

ZF.DDM= 000001            ZF.MAN= 020000            ZF.X3P= 000000            Z.LEN = 000016            $ERR12= ****** GX
ZF.DIA= 004000            ZF.MFL= 000010            ZS.ASN= 100000            Z.LLN   000006            $HEADR= ****** GX
ZF.DLC= 000002            ZF.MTM= 000400            ZS.BSY= 140000            Z.MAP   000020            $NIMFE  000006RG
ZF.DVP= 100000            ZF.MUX= 000040            Z.AVL   000014            Z.NAM   000004            $SAVAL= ****** GX
ZF.INI= 040000            ZF.PSE= 002000            Z.DAT   000016            Z.PCB   000012            $$$ARG= 000004
ZF.KMX= 000020            ZF.SLI= 010000            Z.DSP   000000            Z.SCH   000007            $$$R  = 000010
ZF.LLC= 000004            ZF.TIM= 000200            Z.FLG   000010            $DBECL= ****** GX         .$$$$.= 000034
ZF.LMC= 000100

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        001240    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 113
Work  file writes: 106
Size of work file: 27005 Words  ( 106 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:46.36
DB2:NTIMFE.T47,[132,134]NTIMFE/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTIMFE
```

```
                                                                         M 11
NTIMFE      CREATED BY  MACRO  ON 24-JUL-85 AT 15:09        PAGE 1
```

```
                                         .SBTTL  DLMMFL - DLM MARKED FOR LOAD SCAN

                                 ;+
                                 ;
                                 ; DLMMFL - SCAN DLM CIRCUITS FOR MARKED-FOR-LOAD
                                 ;
                                 ; INPUTS:
                                 ;       R1 - DLM'S SLT ADDRESS
                                 ;       R2 - DLM'S PDV ADDRESS
                                 ;
                                 ; OUTPUTS:
                                 ;       CIRCUITS MARKED-FOR-LOAD ARE LOADED
                                 ;       ALL REGISTERS ARE PRESERVED
                                 ;
                                 ;-
000344  004567  000000G          DLMMFL: JSR     R5,$SAVRG               ; SAVE R3-R5
000350  116104  000014                   MOVB    L.NSTA(R1),R4           ; GET NUMBER OF CIRCUITS
000354  005005                           CLR     R5                      ; INITIALIZE CIRCUIT COUNT
000356  010103                           MOV     R1,R3                   ; POINT TO SLT
000360  062703  000022                   ADD     #L.MPF,R3               ; POINT TO FIRST CIRCUIT
000364  132763  000040  000000   10$:    BITB    #SF.MFL,S.FLG(R3)       ; IS THIS CIRCUIT MARKED-FOR-LOAD?
000372  001424                           BEQ     30$                     ; BR IF NO
000374  016267  000004  000002G          MOV     Z.NAM(R2),$NTLFC+LR.CIR ; STORE PROCESS NAME
000402  012767  004010  000000G          MOV     #LS.CIR+LS.DLM,$NTLFC+LR.STS ; STORE FUNCTION CODE
000410  105767  000001G                  TSTB    $NIFLG+1                ; TOP-DOWN ALLOCATION?
000414  001403                           BEQ     20$                     ; BR IF NO
000416  052767  002000  000000G          BIS     #LS.TOP,$NTLFC+LR.STS   ; MARK THE REQUEST BLOCK
000424  110567  000015G          20$:    MOVB    R5,$NTLFC+LO.STA        ; STORE STATION NUMBER
000430  116167  000012  000014G          MOVB    L.CTL(R1),$NTLFC+LO.CTL ; STORE CONTROLLER NUMBER
000436                                   CALL    $NINTL                  ; REQUEST NTL TO LOAD CIRCUIT
000442  103410                           BCS     40$                     ; BR IF ERROR
000444  142763  000040  000000   30$:    BICB    #SF.MFL,S.FLG(R3)       ; CLEAR MARKED-FOR-LOAD BIT
000452  005205                           INC     R5                      ; UPDATE CIRCUIT COUNT
000454  062703  000004                   ADD     #S.LEN,R3               ; POINT TO NEXT CIRCUIT
000460  005304                           DEC     R4                      ; MORE STATIONS?
000462  003340                           BGT     10$                     ; BR IF YES
000464                           40$:    RETURN

        000001                           .END
```

```
ERRTAB: .WORD   $ERR04,20.              ;COMM EXEC NOT LOADED
        .WORD   $ERR11,21.              ;ROUTER PRI DATA BASE ALLOCATION FAILURE
        .WORD   $ERR15,24.              ;NS: ASSIGNMENT FAILURE
        .WORD   $ERR17,25.              ;NETACP NOT INSTALLED
        .WORD   $ERR19,26.              ;RCP NOT INSTALLED
        .WORD   $ERR20,27.              ;DEVICE OFFLINE
        .WORD   TABEND                  ;END OF TABLE

        TABEND = -1                     ;END OF TABLE CONSTANT DEFINITION

        .IFF

$NIFNC == $NIRDB

        .ENDC   ;DF S$$BAS

        .ENDC   ;NDF R$$11S
```

```
000616  005725                          TST     (R5)+                   ;SKIP ADDRESS OF FORMAT STRING
000620  010504                          MOV     R5,R4                   ;COPY ADDRESS AGAIN
000622  022424                          CMP     (R4)+,(R4)+             ;POINT AT ERROR MESSAGE

000624  112410                  10$:    MOVB    (R4)+,(R0)              ;COPY A BYTE INTO BUFFER
000626  001412                          BEQ     20$                     ;BR IF END OF MESSAGE
000630  122720  000052                  CMPB    #'*,(R0)+               ;FOUND AN ASTERISK?
000634  001373                          BNE     10$                     ;BR IF NO
000636  005715                          TST     (R5)                    ;IS THERE AN ASTERISK REPLACER?
000640  001402                          BEQ     15$                     ;BR IF NO
000642  005300                          DEC     R0                      ;OVERWRITE THE ASTERISK
000644                                  CALL    @(R5)+                  ;CALL THE REPLACER

000646  112420                  15$:    MOVB    (R4)+,(R0)+             ;COPY REST OF MESSAGE
000650  001376                          BNE     15$                     ;UNTIL A NULL BYTE
000652  005300                          DEC     R0                      ;BACK UP TO NULL

000654  010005                  20$:    MOV     R0,R5                   ;COPY ADDRESS OF END OF BUFFER
000656  162705  000230'                 SUB     #NTIERN,R5              ;GET LENGTH OF MESSAGE
000662                                  QIOW$S  #IO.WVB,#1,#2,,,,<#NTIERN,R5,#40>

000732  012605                          MOV     (SP)+,R5                ;RECOVER R5
000734  012604                          MOV     (SP)+,R4                ;RECOVER R4
000736  012603                          MOV     (SP)+,R3                ;RECOVER R3
000740  012602                          MOV     (SP)+,R2                ;RECOVER R2
000742  012601                          MOV     (SP)+,R1                ;RECOVER R1
000744  000261                          SEC
000746                                  RETURN

                                        .ENDC   ;DF S$$BAS


```

```
                                                                 L 15
NTINIT      CREATED BY  MACRO  ON 5-SEP-85 AT 11:04        PAGE 5
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL  VALUE             REFERENCES

ZF.LMC  = 000100          #5-75
ZF.MAN  = 020000          #5-75
ZF.MFL  = 000010          #5-75
ZF.MTM  = 000400          #5-75
ZF.MUX  = 000040          #5-75
ZF.PSE  = 002000          #5-75
ZF.SLI  = 010000          #5-75
ZF.TIM  = 000200          #5-75
ZF.X3P  = 000000          #5-75
ZS.ASN  = 100000          #5-75
ZS.BSY  = 140000          #5-75
Z.AVL     000014          #5-75
Z.DAT     000016          #5-75       12-599     12-609
Z.DSP     000000          #5-75       5-75
Z.FLG     000010          #5-75
Z.LEN   = 000016          #5-75
Z.LLN     000006          #5-75
Z.MAP     000020          #5-75
Z.NAM     000004          #5-75
Z.PCB     000012          #5-75       12-597     12-607
Z.SCH     000007          #5-75
$ABTIM  = ******    GX     7-224
$ALOCB  = ******    GX     7-216
$CALLX  = ******    GX     7-235
$CBDMG  = ******    GX     14-730     14-742
$CCBGT  = ******    GX     7-239
$CEACC  = ******    GX     7-242
$CLKHD  = ******    GX     7-222
$C5TA   = ******    GX     14-716
$DBECL    000004 RG       #6-99      *12-587    *12-599
$DBXPT    000006 RG       #6-100     *12-588    *12-609
$DEACB  = ******    GX     7-217
$DECPT  = ******    GX     7-245
$DEVHD  = ******    GX     7-225
$ERR02    000352 R        #7-190      11-564
$ERR03    000430 R        #7-192      9-378
$ERR04    000462 R        #7-193      9-380
$ERR12    000520 RG       #7-194
$FMASK  = ******    GX     7-203
$HEADR  = ******    GX     7-219      17-862
$LDPWF  = ******    GX     7-226
$LGSTT  = ******    GX     7-244
$LLCAL  = ******    GX     7-248
$LLCRS  = ******    GX     7-240
$LLCTA  = ******    GX     7-249
$NERDD    001014 RG       #14-737
$NERDN    000772 RG       #14-726     14-737
$NER50    000750 RG       #14-715     14-726
$NIAST    000120 R        #6-112     *8-302     *8-305     9-441      17-869
$NICEX  = ******    GX     9-387
$NIECL    000000 RG       #6-97      *12-585    *12-594
$NIERA    000010 RG       #6-102      14-715     14-728    14-738
```

```
                                                                 M 15
NTINIT      CREATED BY  MACRO  ON 5-SEP-85 AT 11:04        PAGE 6
```

```
                                                               L 16
NTITIM      CREATED BY  MACRO  ON 29-JUN-85 AT 03:28      PAGE 1
SYMBOL  CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

ABTIM    = ******   GX       8-178
BUFUMP   = 172354           #5-62
CLINS      000162 R          7-152       #8-173
CLKHD    = ******   GX       6-94         8-182
CMODE    = 140000           #5-62
C.TIM      000006            8-186        8-189
I$$AS    = ******            5-62
KISAR0   = 172340           #5-62
KISAR6   = 172354           #5-62
MPAR     = 172100           #5-62
MPCSR    = 177746           #5-62
M$$MGE   = 000000            5-62
PIRQ     = 177772           #5-62
PMODE    = 030000           #5-62
PR0      = 000000           #5-62
PR1      = 000040           #5-62
PR2      = 000100           #5-62
PR3      = 000140           #5-62
PR4      = 000200           #5-62
PR5      = 000240           #5-62
PR6      = 000300           #5-62
PR7      = 000340           #5-62         6-98
PS       = 177776           #5-62        *6-98        *6-111
R$$MPL   = ******            5-62         5-62         5-62         9-198
R$$11D   = ******            5-62
SWR      = 177570           #5-62
TIFLG      000000 R         #5-72        *6-103       *6-108        6-112        6-115
TKPS     = ******   GX       7-143
TK100    = ******   GX       7-149
TPS      = 177564           #5-62
T1       = 000002           #5-70         6-108        6-115
T1SCL    = ******   GX       6-106        7-142
T1SIN      000116 R          6-117       #7-141
T100C    = ******   GX       6-101        7-148
T100Q    = ******   GX       7-146
T5       = 000001           #5-69         6-103        6-112
T50IN      000130 R          6-114       #7-145
UBMPR    = 170200           #5-62
UISAR0   = 177640           #5-62
UISAR1   = 177642           #5-62
$NITIM     000002 RG        #6-94
.BASEB   = 140000           #5-62
```

```
                                                               M 16
NTITIM      CREATED BY  MACRO  ON 29-JUN-85 AT 03:28      PAGE 2
```

```
                                                                                     N 1
VUNL - VNP UNLOAD ROUTINES      MACRO V05.03b Monday 16-Sep-85 07:30  Page 25-3
Symbol table

$CCBSZ= ****** GX       $FLAGS= ****** GX       $PNAME  000000RG    002 $XLBR = ****** GX       .DEAPR= ****** GX
$CCEX   003666RG        $FMASK= ****** GX       $PRAVL= ****** GX       $XLIN   001320RG        .DEA16  006144R
$CEACC= ****** GX       $GETAD= ****** GX       $PUTAD= ****** GX       $XLINK= ****** GX       .NETPF= ****** GX
$CETDF= ****** GX       $GETRV= ****** GX       $PUTRC= ****** GX       $XPAR = ****** GX       .NS0   = ****** GX
$CLIN   001610RG        $HMBUF= ****** GX       $RADDR= ****** GX       $XPRO   000132RG        .NS1   = ****** GX
$CPRO   001462RG        $ICBLN= ****** GX       $RBLCK= ****** GX       $XPSI = ****** GX       .NS2   = ****** GX
$CXOPT= ****** GX       $KRBLN= ****** GX       $RDBLH= ****** GX       $XSNA = ****** GX       .NS3   = ****** GX
$DDMPV= ****** GX       $LGSTT= ****** GX       $RSIZE= ****** GX       $$     = 177777         .NS4   = ****** GX
$DEA16= ****** GX       $MUL   = ****** GX      $SAVAL= ****** GX       $$$    = 003600R        .NS5   = ****** GX
$DEA18= ****** GX       $MXVCT= ****** GX       $SDBLH= ****** GX       $$$$   = 003664R        .NS6   = ****** GX
$DECPT= ****** GX       $NAME  = ****** GX      $SLTA  = ****** GX      .CXALL= ****** GX       .NS7   = ****** GX
$DEPOL= ****** GX       $NTLHB= ****** GX       $SLTMA= ****** GX       .CXCSR= ****** GX       .PLGTH= ****** GX
$DEUMR= ****** GX       $OBJHD= ****** GX       $SLTNM= ****** GX       .CXLEN= ****** GX       .SBPCB= ****** GX
$DEVHD= ****** GX       $PARHD= ****** GX       $UNBLK= ****** GX       .CXPCB= ****** GX       .TLGTH= ****** GX
$DLCPV= ****** GX       $PDVA  = ****** GX      $UNBSZ= ****** GX       .CXUNL= ****** GX       .TSKHD= ****** GX
$EROUT= ****** GX       $PDVNM= ****** GX       $VFLAG= ****** GX       .CXVEC= ****** GX       ..PDV   000202R      002
$FILHD= ****** GX       $PDVTA= ****** GX       $VUNB  = ****** GX

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        006346    001   (RW,I,LCL,REL,CON)
DATA    000204    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 77
Work  file writes: 76
Size of work file: 21423 Words  ( 84 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:02:09.44
DB2:VUNL.134,[132,134]VUNL/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]VUNL
```

```
                                                                N 1
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31      PAGE 1
```

```
                                        .SBTTL  MACRO DEFINITIONS

                                ;***
                                ; LIBRARY MACROS
                                ;***
                                        .MCALL  PVCDF$,CUGDF$,RDTDF$,DSTDF$,DTEDF$,XCBDF$,PHBDF$,RNWDF$         ;[TD001]
                                        .MCALL  CHNDF$,SVCDF$,X29DF$,TRADF$,ASL$,SAVRG,RESRG,TCBDF$             ;[TD001]
                                        .MCALL  GETRV,PUTRC,GETAD,PUTAD,CEACC$,SWTCHI,SWTCHO                    ;[TD001]
                                        .MCALL  VNPDBG                                                          ;[TD001]
                                                                                                                ;**-3
 000000                                 CHNDF$                          ; DEFINE OUTGOING CHANNELS LIST OFFSETS
 000000                                 CUGDF$                          ; DEFINE CLOSED USER GROUP BLOCK OFFSETS
 000000                                 DSTDF$                          ; DEFINE X.25/X.29 DESTINATION BLOCK OFFSETS
 000000                                 DTEDF$                          ; DEFINE LOCAL DTE DESCRIPTOR OFFSETS
 000000                                 PHBDF$                          ; DEFINE X.25 HOME BLOCK OFFSETS
 000000                                 PVCDF$                          ; DEFINE PVC OFFSETS
 000000                                 RDTDF$                          ; DEFINE REMOTE DTE BLOCK OFFSETS
 000000                                 SVCDF$                          ; DEFINE SVC BLOCK OFFSETS
 000000                                 XCBDF$                          ; DEFINE X.25 CIRCUIT BLOCK OFFSETS
 000000                                 X29DF$                          ; DEFINE X.29 DATA BLOCK OFFSETS
 000000                                 TCBDF$                          ; DEFINE TCB DATA STRUCTURE OFFSETS
 000000                                 TRADF$                          ; DEFINE TRACE DATA STRUCTURE OFFSETS
 000000                                 RNWDF$                          ; Define remote network block symbols   ;[TD001]
                                ;
                                ; LOCAL MACRO DEFINITIONS
                                ;
                                        .MACRO  SAVMAP
                                        MOV     $RBLCK,-(SP)
                                        .ENDM

                                        .MACRO  RESMAP
                                        MOV     (SP)+,$RBLCK
                                        .ENDM
```

```
                                                  .SBTTL  XPSV - Deallocate PSV data structures            ;[TD001]
                                          ;+                                                               ;[TD001]
                                          ; *** XPSV - Deallocate PSV data structures                      ;[TD001]
                                          ;                                                                ;[TD001]
                                          ; INPUTS                                                         ;[TD001]
                                          ;       $PFLAG - PSI characteristics word                        ;[TD001]
                                          ;                                                                ;[TD001]
                                          ; OUTPUTS                                                        ;[TD001]
                                          ;       PSV mapping tables deallocated                           ;[TD001]
                                          ;       All registers preserved                                  ;[TD001]
                                          ;-                                                               ;[TD001]
001720  032767  000000G 000000G   XPSV:   BIT     #PF.PSV,$PFLAG  ; Check if PSV is present                ;[TD001]
001726  001446                            BEQ     50$             ; Nothing to do if not                   ;[TD001]
                                          ;                                                                ;[TD001]
001730                                    CALL    $SAVAL          ; Save all registers                     ;[TD001]
001734  016703  000000G                   MOV     $PSIPT,R3       ; Get address of pointer                 ;[TD001]
001740  062703  000044                    ADD     #H$GLT,R3       ; ... to first table                     ;[TD001]
001744                                    GETRV   R3,#2           ; Read in pointer                        ;[TD001]
001762  016700  000000G                   MOV     $BFR,R0         ; Get pointer                            ;[TD001]
001766  001404                            BEQ     20$             ; Branch if no table allocated           ;[TD001]
001770  012701  001000                    MOV     #512.,R1        ; Get length of table                    ;[TD001]
001774                                    CALL    $DEPOL          ; Deallocate table                       ;[TD001]
002000  016703  000000G           20$:    MOV     $PSIPT,R3       ; Get address of pointer                 ;[TD001]
002004  062703  000046                    ADD     #H$GPT,R3       ; ... to second table                    ;[TD001]
002010                                    GETRV   R3,#2           ; Read in pointer                        ;[TD001]
002026  016700  000000G                   MOV     $BFR,R0         ; Get pointer                            ;[TD001]
002032  001404                            BEQ     50$             ; Branch if no table allocated           ;[TD001]
002034  012701  001000                    MOV     #512.,R1        ; Get length of table                    ;[TD001]
002040                                    CALL    $DEPOL          ; Deallocate table\                      ;[TD001]
                                          ;                                                                ;[TD001]
002044                            50$:    RETURN                                                           ;[TD001]
```

```
243 000364  001002                             BNE     30$                  ; IF NE, NO
244 000366  010162  000000G                    MOV     R1,L.DLS(R2)         ; SET DLC LINE TABLE ADDRESS
245 000372  166701  000000G            30$:    SUB     $MXLEN,R1            ; GET NEXT UNIT PORTION ADDRESS
246 000376  005316                     40$:    DEC     (SP)                 ; ANY MORE ENTRIES LEFT ?
247 000400  003353                             BGT     20$                  ; IF GT, YES
248 000402  005726                             TST     (SP)+                ; ELSE, CLEAN OFF STACK
249 000404  000410                             BR      70$                  ; DON'T SET DDM LINE TABLE ADDRESS AGAIN !
250 000406  010160  000000G            50$:    MOV     R1,L.DDS(R0)         ; SET DDM DATA BASE ADDRESS
251 000412  126060  000000G 000000G    55$:    CMPB    L.DDM(R0),L.DLC(R0)  ; COMBINATION DDM/DLC?
252 000420  001002                             BNE     70$                  ; IF NE, NO
253 000422  010160  000000G            60$:    MOV     R1,L.DLS(R0)         ; SET DLC DATA BASE ADDRESS
254 000426                             70$:    CALL    $TMRDY               ; MARK LINE READY IF APPROPRIATE
255 000432                             80$:    RETURN
```

```
VXIT        CREATED BY  MACRO  ON 29-JUN-85 AT 03:12      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL    VALUE               REFERENCES

D$END     = 000104         5-150     5-155
D$LNAM      000006         5-151
ERR1        000426 R       5-197    #5-200
FE.CEX    = ****** GX      5-161
F.ERR     = ****** GX      5-200
F.FNB     = ****** GX       5-85
F.NRBD    = ****** GX     *5-106    *5-108    *5-110    *5-112    *5-146
F.RSIZ    = ****** GX     *5-107    *5-150
F.URBD    = ****** GX     *5-145
L$BBLK    = ****** GX      5-170
L$BXFR    = ****** GX     *5-175
N.FID     = ****** GX       5-85
PAR$$$    = 000027        #5-186     5-186
R$$MPL    = ******          5-96     5-115     5-174
SYSFDB    = ****** GX       5-85    *5-106    *5-107    *5-108    *5-110    *5-112    *5-145    *5-146    *5-150
                           5-186     5-186     5-200
VF.INI    = ****** GX       5-89
V1.NSY    = ****** GX      5-167
$BFR      = ****** GX      5-145     5-146     5-151    *5-175
$CRAVL    = ****** GX      5-112
$DEA16    = ****** GX      5-156
$DECPT    = ****** GX      5-148     5-150     5-154    *5-157
$DEVHD    = ****** GX      5-108
$EROUT    = ****** GX      5-197
$FMASK    = ****** GX      5-106     5-161
$GETRC    = ****** GX      5-172
$GETRV    = ****** GX      5-150
$PARHD    = ****** GX      5-110
$PUTRC    = ****** GX      5-107     5-109     5-111     5-113
$PUTRD    = ****** GX      5-180
$RADDR    = ****** GX     *5-107    *5-109    *5-111    *5-113    *5-150    *5-169
$RBLCK    = ****** GX     *5-171
$RSIZ     = ****** GX      5-184
$RSIZE    = ****** GX     *5-170    *5-184
$VFLAG    = ****** GX       5-89
$VFLG1    = ****** GX      5-167
$VIMO     = ****** GX      5-163
$VXIT       000000 RG      #5-80
$XFRAD    = ****** GX      5-175
$$        = 177777         5-199     5-199    #5-199    #5-200
$$$       = 000354 R      #5-199     5-200
$$$$      = 000426 R      #5-200
.CLOSE    = ****** G       5-186     5-186
.CRAVL    = ****** GX      5-113
.DEVHD    = ****** GX      5-109
.FMASK    = ****** GX      5-107
.PARHD    = ****** GX      5-111
```

```
VXIT        CREATED BY  MACRO  ON 29-JUN-85 AT 03:12      PAGE 2
```

```
                                      ;+
                                      ; $XPAR - RELEASE MCB PROCESS FROM PARTITION
                                      ;
                                      ; THIS ROUTINE IS CALLED TO FREE THE PROCESS PARTITION. IF IT IS A SYSTEM
                                      ; CONTROLLED PARTITION, IT WILL BE UNLINKED FROM THE LIST OF SUB-PCB'S
                                      ; AND DE-ALLOCATED.
                                      ;
                                      ; INPUTS:
                                      ;       R0=PCB ADDRESS
                                      ;
                                      ; OUTPUTS:
                                      ;       C-BIT=SUCCESS/FAILURE
                                      ;-

                                               .MCALL   PCBDF$,GETRV,PUTRC,HWDDF$,SWTCHO,SWTCHI
000000                                         PCBDF$                          ; PCB OFFSETS
000000                                         HWDDF$                          ; HARDWARE DEFS

000000                                $XPAR::
000000                                         GETRV    R0,.PLGTH              ; READ IN PCB
000016    016701   000012G                     MOV      P.MAIN+$BFR,R1         ; GET MAIN PARTITION PCB ADDRESS
000022    016767   000010G 000060G             MOV      P.SUB+$BFR,$BFR+60     ; SAVE NEXT PCB ADDRESS
000030                                         SWTCHI   #$BFR+100              ; PREPARE TO RECEIVE MAIN PCB
000036                                         GETRV    R1,.PLGTH              ; READ IN MAIN PCB
000054                                         SWTCHI                          ; DEFAULT BUFFER

                                               .IF      NDF,R$$MPL

000062    032767   000040  000130G            BIT      #PS.SYS,P.STAT+$BFR+100 ; SYSTEM CONTROLLED MAIN PARTITION?
000070    001035                               BNE      10$                    ; IF NE, YES

000072    146767   000024G 000025G 5$:         BICB     P.BUSY+$BFR,P.BUSY+$BFR+1       ; UN-BUSY PARTITION
000100    146767   000024G 000125G             BICB     P.BUSY+$BFR,P.BUSY+$BFR+101     ; UN-BUSY MAIN PARTITION

000106    032767   000020  000000G             BIT      #FE.PLA,$FMASK         ; PLAS SYSTEM
000114    001406                               BEQ      7$                     ; NO IF EQ
000116    010146                               MOV      R1,-(SP)               ; SAVE R1
000120    016701   000000G                     MOV      .PPRO,R1               ; P.PRO OFFSET IN PCB
000124    005061   000000G                     CLR      $BFR(R1)               ; CLEAR THE PROTECTION WORD
000130    012601                               MOV      (SP)+,R1               ; RESTORE R1

000132                                7$:      PUTRC                           ; RESTORE PCB
000136                                         SWTCHO   #$BFR+100              ; SWTCH OUTPUT BUFFERS
000144                                         PUTRC    R1                     ; WRITE MAIN PCB
000154                                         SWTCHO                          ; DEFAULT BUFFER
000162    000427                               BR       40$

                                               .ENDC

000164    020001                      10$:     CMP      R0,R1                  ; IS THIS A SUB-PCB?
000166    001425                               BEQ      40$                    ; IF EQ, NO
000170                                20$:     GETRV    R1                     ; READ IN PCB
000200    026700   000010G                     CMP      P.SUB+$BFR,R0          ; DOES THIS SUB-PCB LINK TO OURS?
000204    001405                               BEQ      30$                    ; IF EQ, YES
000206    016701   000010G                     MOV      P.SUB+$BFR,R1          ; GET NEXT SUB-PCB
000212    001364                               BNE      10$                    ; IF NE, GOT ONE
000214    000261                               SEC                             ; SIGNAL FAILURE
```

```
    274                                                                         ; UNTIL NEXT BASE LEVEL         ;[MP01]
    275                                         CALL    $NXHCK                  ; INIT DLX IF NECESSARY         ;[MP01]
    276
    277                                         .ENDC                                                           ;[MP01]
    278
    279 000302                                  CALL    $NITIM                  ;GO START TIMERS
    280 000306  000241                          CLC                             ;GOOD FINISH
    281 000310                          40$:    RETURN
    282
    283
    284
```

```
153 000146  103325                          BCC     10$                     ;BR IF NO, CHECK NEXT
154                                 ;
155                                 ; FOUND A BAD CSR
156                                 ;
157 000150  016700  000016'                 MOV     CSRSLT,R0               ;PICK UP SLT ADDRESS
158
159                                         .IF NDF S$$BAS
160
161 000154                                  SAVRG   <R1>                    ;SAVE R1
162 000156  116001  000002                  MOVB    L.DDM(R0),R1            ;GET DDM PDV INDEX
163 000162  067701  000000G                 ADD     @PDVTA,R1               ;INDEX INTO PDV VECTOR TABLE
164 000166  011101                          MOV     (R1),R1                 ;GET PDV ADDRESS
165 000170  016101  000004                  MOV     Z.NAM(R1),R1            ;GET DDM NAME
166
167                                         .ENDC   ;NDF S$$BAS
168
169                                         .IFDF   R$$11S
170
171                                         BIC     #LF.RDY!LF.ENA,L.FLG(R0);INDICATE LINE IS UNUSABLE
172
173                                         .IFF    ;DF R$$11S
174
175 000174  012767  000000G 000000C         MOV     #LX.LIN,$NTLFC+LR.STS   ;SET ''CLEAR LINE'' FUNCTION CODE
176 000202  116067  000012  000000C         MOVB    L.CTL(R0),LR.CTL+$NTLFC ;SET CONTROLLER NUMBER
177 000210  116067  000013  000000C         MOVB    L.UNT(R0),LR.UNT+$NTLFC ;SET UNIT NUMBER
178 000216  010167  000000C                 MOV     R1,LR.LIN+$NTLFC        ;GET DDM NAME (RAD50)
179 000222                                  CALL    $NINTL                  ;SEND REQUEST TO NTL
180
181                                         .IFTF   ;DF R$$11S
182
183                                                 .IF NDF S$$BAS
184
185 000226  005067  000002G                 CLR     $NIERA+2                ;MAKE A NULL
186 000232  116067  000012  000002G         MOVB    L.CTL(R0),$NIERA+2      ;SET CONTROLLER NUMBER
187 000240  116000  000002                  MOVB    L.DDM(R0),R0            ;GET DDM PDV INDEX (SIGN BIT NEVER ON)
188 000244  010167  000000G                 MOV     R1,$NIERA               ;SET RAD50 DEVICE/PDV NAME
189 000250                                  EMSG$   20                      ;DEVICE ''DDM-NUM'' OFFLINE
190 000256                                  RESRG   <R1>                    ;RESTORE R1
191
192                                                 .ENDC   ;NDF S$$BAS
193
194                                         .ENDC   ;DF R$$11S
195 000260  000660                          BR      10$                     ;CHECK REST ANYWAY
196
197 000262  000241                  50$:    CLC                             ;ALWAYS RETURN SUCCESS
198 000264                                  RESRG   <R2,R1,R0>              ;RESTORE REGISTERS
199 000272                          60$:    RETURN
200
201                                 ;
202                                 ; ILLEGAL ADDRESS TRAP ROUTINE, EXPECTS A ZERO ENTRY ON TOP OF USER STACK
203                                 ;
204 000274  005266  000004          CSRTRP: INC     4(SP)                   ;SET TOP OF USER STACK TO A ''ONE''
205 000300  000002                          RTI                             ;RETURN WITH ''BAD CSR'' INDICATION
206
207
208         000001                          .END
```

```
 000132  032701  000100           BIT     #NF.LV2,R1             ;* LEVEL 2 ROUTER ?
 000136  001404                   BEQ     40$                    ;* IF NO - BRANCH

 000140  012703  000010'          MOV     #RCP2NM,R3             ;* POINT TO TASK NAME
 000144                           CALL    @SRSTD                 ;* IS LEVEL 2 ROUTING CONTROL PR
 000150  006166  000004   40$:    ROL     4(SP)                  ;* SAVE CARRY ...
 000154  006366  000004           ASL     4(SP)                  ;* IN 2ND BIT
 000160                   50$:    RETURN                         ;* LEAVE SYSTEM STATE

 000162  006001           60$:    ROR     R1                     ; SET CARRY IF NS IS NOT INSTALLED
 000164  103003                   BCC     70$                    ; IF NOT SET - CONTINUE
 000166                           EMSG$R  15                     ; NS.... NOT INSTALLED
 000174  006001           70$:    ROR     R1                     ; SET CARRY IF RCP(1 OR 2) IS NOT INSTALLED
 000176  103003                   BCC     80$                    ; IF NOT SET - CONTINUE
 000200                           EMSG$R  19                     ; RCP(1 OR 2) NOT INSTALLED
 000206  016702  000000G  80$:    MOV     $DBXPT,R2              ; GET ADDRESS OF XPT'S DDB

 000212                   90$:    .IF NDF R$$11S

 000212  005762  000020           TST     N$PLD+2(R2)            ; IS THERE A PLD DATA BASE ?
 000216  001067                   BNE     150$                   ; IF NE, YES
 000220  017701  000000G          MOV     @DECPT,R1              ; GET DECNET HOME BLOCK ADDRESS
 000224  016100  000030           MOV     D$NLN(R1),R0           ; NUMBER OF NSP CHANNELS
 000230  010062  000016           MOV     R0,N$PLD(R2)           ; STORE # OF CHANNELS IN NSP'S DB
 000234  012701  000052           MOV     #P$LEN,R1              ; GET SIZE OF EACH PLD
 000240                           CALL    $MUL                   ; CALCULATE SIZE OF PLD'S
 000244  016200  000016           MOV     N$PLD(R2),R0           ; GET NUMBER OF CHANNELS * 2
 000250  006300                   ASL     R0                     ; ...
 000252  060001                   ADD     R0,R1                  ; UPDATE NUMBER OF BYTES NEEDED
 000254  005000                   CLR     R0                     ; ASSUME FAILURE

 000256                           SWSTK$  120$                   ;* ENTER KERNEL MODE
 000262                           CALL    @ALOCB                 ;* TRY TO ALLOCATE THE PLD DB
 000266  103417                   BCS     110$                   ;* IF CS, FAILURE
 000270  010066  000002           MOV     R0,2(SP)               ;* ELSE, RETURN ADDRESS IN USER R0
 000274  017701  000000G          MOV     @DECPT,R1              ;* GET DECNET HOME BLOCK ADDRESS
 000300  016104  000030           MOV     D$NLN(R1),R4           ;* GET NUMBER OF CHANNELS
 000304  010401                   MOV     R4,R1                  ;* GET NUMBER OF CHANNELS * 2
 000306  006301                   ASL     R1                     ;* ...
 000310  010003                   MOV     R0,R3                  ;* POINT TO PLD ALLOCATION
 000312  060100                   ADD     R1,R0                  ;* POINT PAST PLD VECTOR
 000314  010023          100$:    MOV     R0,(R3)+               ;* SET UP PLD VECTOR
 000316  062700  000052           ADD     #P$LEN,R0              ;* POINT TO NEXT ENTRY IN PLD
 000322  005304                   DEC     R4                     ;* ALL CHANNELS PROCESSED?
 000324  003373                   BGT     100$                   ;* BR IF NO

 000326                  110$:    RETURN                         ;* BACK TO USER MODE

 000330  010062  000020  120$:    MOV     R0,N$PLD+2(R2)         ; SAVE PLD ADDRESS
 000334  001003                   BNE     130$                   ; IF SUCCESFUL ALLOCATION - BRANCH
 000336                           EMSG$R  18                     ; PLB ALLOCATION FAILURE

 000344  010067  000000'  130$:   MOV     R0,PLDADR              ; SAVE PLB ADDRESS AGAIN
 000350  010167  000002'          MOV     R1,PLDLEN              ; AND LENGTH OF THE PLD DATA BASE
 000354  006201                   ASR     R1                     ; CONVERT BYTES TO WORDS
 000356  016246  000016           MOV     N$PLD(R2),-(SP)        ; GET NUMBER OF CHANNELS * 2
 000362  161601                   SUB     (SP),R1                ; DON'T ZERO PLD VECTOR
```

```
**FILE**ID**NTIMFE
```

```
NN      NN   TTTTTTTTTT     IIIIII     MM      MM   FFFFFFFFFF   EEEEEEEEEE
NN      NN   TTTTTTTTTT     IIIIII     MM      MM   FFFFFFFFFF   EEEEEEEEEE
NN      NN       TT           II       MMMM  MMMM   FF           EE
NN      NN       TT           II       MMMM  MMMM   FF           EE
NNNN    NN       TT           II       MM  MM  MM   FF           EE
NNNN    NN       TT           II       MM  MM  MM   FF           EE
NN  NN  NN       TT           II       MM      MM   FFFFFFFF     EEEEEEEE
NN  NN  NN       TT           II       MM      MM   FFFFFFFF     EEEEEEEE
NN    NNNN       TT           II       MM      MM   FF           EE
NN    NNNN       TT           II       MM      MM   FF           EE
NN      NN       TT           II       MM      MM   FF           EE               ....
NN      NN       TT           II       MM      MM   FF           EE               ....
NN      NN       TT         IIIIII     MM      MM   FF           EEEEEEEEEE       ....
NN      NN       TT         IIIIII     MM      MM   FF           EEEEEEEEEE       ....
```

```
LL               SSSSSSSS   TTTTTTTTTT
LL               SSSSSSSS   TTTTTTTTTT
LL             SS               TT
LL             SS               TT
LL             SS               TT
LL             SS               TT
LL               SSSSSS         TT
LL               SSSSSS         TT
LL                     SS       TT
LL                     SS       TT
LL                     SS       TT
LL                     SS       TT
LLLLLLLLLL     SSSSSSSS         TT
LLLLLLLLLL     SSSSSSSS         TT
```

```
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:08
Table of contents
```

```
                                                                 M 11
NTIMFE     CREATED BY  MACRO  ON 24-JUL-85 AT 15:09     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE              REFERENCES

ACTCIR    000004 R         #5-93      *6-119     6-190     *6-196     6-234     *6-240
CCBGT   = ******  GX        7-289
CEACC   = ******  GX        8-364
CTLNUM    000003 R         #5-92      *6-166     8-366
C.BUF1    000014           *7-291
C.FNC     000010           *7-296
C.LIN     000006           *7-293
C.NSP     000004           *7-294
C.STA     000007           *7-292
C.STS     000012           *7-295
DECPT   = ******  GX        6-187      6-231
DLMSET    000772 R          7-282     #8-328
D$NN      000060            6-188      6-232
ENAC      000074 R         #6-123      6-249
ENALIN    000610 R          6-198      6-241    #7-269
E$NBR     000014           #5-72
E$NBS     000020           #5-72
E$NCR     000034           #5-72
E$NCS     000036           #5-72
E$NIC     000044           #5-72
E$NLEN    000050           #5-72
E$NLLA    000012           #5-72
E$NLNK    000000           #5-72
E$NML     000040           #5-72
E$NMR     000024           #5-72
E$NMS     000030           #5-72
E$NNOD    000002           #5-72
E$NRT     000042           #5-72
E$NRTP    000005           #5-72
E$NSEG    000010           #5-72
E$NTIM    000046           #5-72
E$NUSE    000004           #5-72
E$STRT    000006           #5-72
FC.CTL  = 000006            7-296
FNDPDV  = ******  GX        6-118
FS.STR  = 001000            7-296
H$SVC     000036            8-360
I$$AS   = ******            5-77
KISAR6  = ******  GX        8-342     *8-344     8-359     *8-371     *8-374
KSAR6   = ******  GX        8-370
LF.ACT  = 100000           #5-74       6-132     6-200      6-243
LF.BRO  = 000400           #5-74       6-135
LF.BWT  = 000007           #5-74
LF.ENA  = 002000           #5-74       6-214     6-236
LF.LPB  = 001000           #5-74
LF.MDC  = 000100           #5-74
LF.MFL  = 004000           #5-74
LF.MTP  = 000020           #5-74
LF.PAC  = 000200           #5-74
LF.RDY  = 040000           #5-74       6-125
LF.REA  = 010000           #5-74
LF.SER  = 000040           #5-74
```

```
                                                                 M 11
NTIMFE     CREATED BY  MACRO  ON 24-JUL-85 AT 15:09     PAGE 2
```

```
                                                                M 12
NTIMFL  MACRO V05.03b Saturday 29-Jun-85 03:27  Page 9-1
Symbol table

A$$CHK= 000000        LN.DUM= 000005        LS.OPT= 000400        N$$PEM= 000001            UISAR0= 177640
A$$CPS= 000000        LN.LDA= 000004        LS.PRO= 000002        PDVFLG  000000R       002 UISAR1= 177642
A$$PRI= 000000        LN.LDO= 000003        LS.PWF= 040000        PDVNM  = ****** GX        V$$CTR= 001000
A$$TRP= 000000        LN.DAU= 000003        LS.TOP= 002000        PDVTA  = ****** GX        X$$DBT= 000000
BUFUMP= 172354        LN.OFF= 000001        LS.UNF= 020000        PIRQ   = 177772           ZF.CDU= 001000
CMODE  = 140000       LN.ON  = 000000       LS.X25= 000012        PMODE  = 030000           ZF.DDM= 000001
C$$CKP= 000000        LN.DOP= 000004        LS.11D= 004000        PR0    = 000000           ZF.DIA= 004000
C$$ORE= 000400        LN.OPE= 000001        LX.CEX= 000004        PR1    = 000040           ZF.DLC= 000002
C$$RSH= 177564        LN.REF= 000002        LX.LIN= 000006        PR2    = 000100           ZF.DVP= 100000
DLMMFL  000344R       LN.SER= 000002        LX.PRO= 000005        PR3    = 000140           ZF.INI= 040000
D$$BUG= 177514        LN.STA= 000017        L$$ASG= 000000        PR4    = 000200           ZF.KMX= 000020
D$$ISK= 000000        LN.SUB= 000360        L$$DRV= 000000        PR5    = 000240           ZF.LLC= 000004
D$$L11= 000001        LN.TRI= 000006        L$$P11= 000001        PR6    = 000300           ZF.LMC= 000100
D$$YNC= 000000        LO.CON  000006        L$$11R= 000000        PR7    = 000340           ZF.MAN= 020000
D$$YNM= 000000        LO.CSR  000006        L.COST  000015        PS     = 177776           ZF.MFL= 000010
E$$XPR= 000000        LO.CTL  000014        L.CTL   000012        P$$P45= 000000            ZF.MTM= 000400
F$$LVL= 000001        LO.LIN  000004        L.CVA   177776        P$$WRD= 000000            ZF.MUX= 000040
G$$TPP= 000000        LO.PAR  000010        L.DDM   000002        Q$$DPT= 000010            ZF.PSE= 002000
G$$TSS= 000000        LO.PRI  000010        L.DDS   000004        R$$DER= 000000            ZF.SLI= 010000
G$$TTK= 000000        LO.SCR  000014        L.DLC   000003        R$$K11= 000001            ZF.TIM= 000200
G$$WRD= 000000        LO.STA  000015        L.DLM   000006        R$$SND= 000000            ZF.X3P= 000000
I$$RAR= 000000        LO.VCT  000012        L.DLS   000010        R$$11M= 000000            ZS.ASN= 100000
I$$RDN= 000000        LR.AST  000026        L.FLG   000000        SF.ACT= 000200            ZS.BSY= 140000
KISAR0= 172340        LR.CIR  000002        L.KRBA  000016        SF.ENA= 000100            Z.AVL   000014
KISAR6= 172354        LR.CTL  000004        L.LEN = 000022        SF.LPB= 000004            Z.DAT   000016
K$$CNT= 177546        LR.EFN  000004        L.MPF   000022        SF.MFL= 000040            Z.DSP   000000
K$$CSR= 177546        LR.LIN  000002        L.NMST  000020        SF.PAC= 000020            Z.FLG   000010
K$$LDC= 000000        LR.PRM  000030        L.NSTA  000014        SF.REA= 000010            Z.LEN = 000016
K$$TPS= 000074        LR.PRD  000002        L.OWNR  000021        SF.SER= 000001            Z.LLN   000006
LD$LP  = 000000       LR.STS  000000        L.UNT   000013        SF.SVC= 000002            Z.MAP   000020
LF.ACT= 100000        LR.TCB  000002        MPAR   = 172100       SF.UNL= 000040            Z.NAM   000004
LF.BRO= 000400        LR.UNT  000005        MPCSR  = 177746       SLTMA  = ****** GX        Z.PCB   000012
LF.BWT= 000007        LS.AST= 100000        M$$CRB= 000124        SLTNM  = ****** GX        Z.SCH   000007
LF.ENA= 002000        LS.CEX= 000001        M$$CRX= 000000        SWR    = 177570           $ERR30  000002R       002
LF.LPB= 001000        LS.CIR= 000010        M$$FCS= 000000        S$$WRG= 000000            $ERR31  000042R       002
LF.MDC= 000100        LS.CXO= 010000        M$$MGE= 000000        S$$YSZ= 007600            $NERDD= ****** GX
LF.MFL= 004000        LS.CX5= 000400        M$$NET= 000000        S.COST  000001            $NER50= ****** GX
LF.MTP= 000020        LS.DLM= 004000        M$$OVR= 000000        S.FLG   000000            $NIERA= ****** GX
LF.PAC= 000200        LS.ECH= 001000        N$$ACC= 000001        S.LEN   000004            $NIERR= ****** GX
LF.RDY- 040000        LS.FDX= 004000        N$$BUF= 000001        S.NMST  000002            $NIFLG= ****** GX
LF.RF  = 010000       LS.HDX= 010000        N$$LDV= 000001        S.OWNR  000003            $NIMFL  000000RG
LF.SLR= 000040        LS.LIN= 000003        N$$MCP= 000001        TPS    = 177564           $NINTL= ****** GX
LF.TIM= 000010        LS.LMC= 000007        N$$MLL= 000001        T$$KMG= 000000            $NTLFC= ****** GX
LF.UNL= 020000        LS.NTI= 004000        N$$MOV= 000010        T$$MIN= 000000            $SAVRG= ****** GX
LF.X2P= 000000        LS.ON  = 000011       N$$NCT= 000001        UBMPR  = 170200           .BASEB= 140000
LN.CLO= 000000

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000466    001   (RW,I,LCL,REL,CON)
DATA    000104    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
```

```
                                                                N 12
NTIMFL  MACRO V05.03b Saturday 29-Jun-85 03:27  Page 9-2
Symbol table
```

```
                                         .SBTTL  ERROR MESSAGES

                                ;
                                ; ERROR MESSAGE BUFFER
                                ;
000230     015      116      124  NTIERN: .ASCII  <15>/NTINIT -- /       ;MESSAGE PREFIX (NAME OF NTINIT)
000233     111      116      111
000236     124      040      055
000241     055      040
000243                            NTIERB: .BLKB   70.                    ;MESSAGE BODY
                                          .EVEN

                                ;
                                ; ERROR MESSAGES
                                ;
                                ; MESSAGES IN THIS MODULE SHOULD BE NUMBERED 01 TO 09
                                ; MESSAGES 10-11 ARE IN NTICEX
                                ; MESSAGES 11-19 ARE IN NTILON
                                ; MESSAGES 20-24 ARE IN NTICSR
                                ; MESSAGES 25-29 ARE IN NTIKPN OR NTIKLC
                                ; MESSAGES 30-34 ARE IN NTIMFL
                                ; MESSAGES 35-39 ARE IN NTIMFE
                                ; MESSAGES 40-44 ARE IN NTIX25
                                ;
                                ; THE MAIN ERROR PROCESSOR IS "$NIERR"
                                ; ASTERISK REPLACERS ARE:
                                ;   $NER50 - INSERT RAD50 STRING FROM $NIERA
                                ;   $NERDN - INSERT RAD50 STRING FROM $NIERA AND DECIMAL NUMBER FROM $NIERA+2
                                ;   $NERDD - INSERT RAD50 STRING FROM $NIERA, DECIMAL NUMBER FROM $NIERA+2,
                                ;                  AND DECIMAL NUMBER FROM $NIERA+4
                                ; SECOND LINE FORMATTING AND PRINTING IS NOT SUPPORTED
                                ;
                                          .ENABL  LC
                                          .IF NDF R$$11S
000352                          NTLER$    ,02,NONE,$NIERR,,,<Network Loader (NTL...) Not Installed>
                                          .ENDC
000430                          NTLER$    ,03,NONE,$NIERR,,,<Illegal Function>
000462                          NTLER$    ,04,NONE,$NIERR,,,<Comm Exec Not Loaded>
000520                          NTLER$    <:>,12,NONE,$NIERR,,,<Line in Wrong State, Only First Line Enabled>

                                          .DSABL  LC

                                ;+
                                ; VECTORED EXEC TABLE
                                ;

000606  000000                  EXEVEC: .WORD   0
000610  000000G                 FMASK:: $FMASK                   ; RSX-11M+ VECTORED SYMBOLS
000612  172354                  KSAR6:: KISAR6

                                          .IF DF  R$$MPL
                                HFMSK:: $HFMSK
                                SAHDB:: $SAHDB
                                SAHPT:: $SAHPT
                                URMST:: $URMST
                                NCPU::  $NCPU
                                          .ENDC
```

```
                                  ;       .SBTTL  ERROR SUBROUTINES

                                  ;+
                                  ; $NER50 - INSERT A RAD50 STRING
                                  ; $NERDN - INSERT A RAD50 STRING AND A DECIMAL NUMBER
                                  ; $NERDD - INSERT A RAD50 STRING AND TWO DECIMAL NUMBERS
                                  ;-

                                          .IF NDF S$$BAS
                                  ;
                                  ; $NIERA          = RAD50 WORD
                                  ;
 000750  016701  000010'          $NER50::MOV     $NIERA,R1               ;GET RAD50 NAME
 000754                                   CALL    $C5TA                   ;INSERT RAD50
 000760  122740  000040           10$:    CMPB    #' ,-(R0)               ;SKIP TRAILING BLANKS
 000764  001775                           BEQ     10$                     ;
 000766  005200                           INC     R0                      ;ADJUST FOR AUTO-DECREMENT
 000770                                   RETURN

                                  ;
                                  ; $NIERA          = RAD50 WORD
                                  ; $NIERA+2        = DECIMAL NUMBER
                                  ;
 000772                           $NERDN::CALL    $NER50                  ;INSERT RAD50 STRING
 000776  112720  000055                   MOVB    #'-,(R0)+               ;PUT IN THE SEPARATOR
 001002  016701  000012'                  MOV     $NIERA+2,R1             ;GET OCTAL NUMBER
 001006  005002                           CLR     R2                      ;ZERO SUPPRESS
 001010                                   CALLR   $CBDMG                  ;PUT IN DECIMAL NUMBER

                                  ;
                                  ; $NIERA          = RAD50 WORD
                                  ; $NIERA+2        = DECIMAL NUMBER
                                  ; $NIERA+4        = DECIMAL NUMBER (IF MI, DON'T PRINT)
                                  ;
 001014                           $NERDD::CALL    $NERDN                  ;PUT IN RAD50 AND FIRST OCTAL
 001020  016701  000014'                  MOV     $NIERA+4,R1             ;GET SECOND NUMBER
 001024  100405                           BMI     10$                     ;IF MI, DON'T PRINT
 001026  112720  000055                   MOVB    #'-,(R0)+               ;PUT IN SECOND SEPARATOR
 001032  005002                           CLR     R2                      ;ZERO SUPPRESS
 001034                                   CALLR   $CBDMG                  ;PUT IN DECIMAL NUMBER
 001040                           10$:    RETURN

                                          .ENDC
```

```
                                                                                      M 15
NTINIT      CREATED BY  MACRO  ON 5-SEP-85 AT 11:04          PAGE 6
SYMBOL CROSS REFERENCE                                       CREF    04.00

SYMBOL   VALUE              REFERENCES

$NIERR     000576 RG         7-190      7-192      7-193      7-194     #13-636
$NIEVC     000102 RG        #6-110      9-403      9-427
$NIFLG     000226 RG        #6-127     *9-440     16-799    *16-825
$NIFNC     000204 RG         6-122     #6-123      8-303      9-361      9-367      9-369      9-389      9-393      9-415
                            *9-434     *9-436    *16-822    *16-823    *16-824     17-872
$NILON   = ******   GX       9-404
$NIMRK     000070 R         #6-109      9-401      9-425
$NINTL     000376 RG       #11-556
$NIPRM     000122 R         #6-113     *8-306     17-871
$NIRDB     000032 RG        #6-106      8-305      8-306
$NIRNL     000134 FG        #6-118     11-559
$NIRTN     000026 RG         6-104     #6-105      6-116     16-798
$NISNL     000152 RG        #6-119     11-557
$NIXPT     000002 R(        #6-98     *12-586    *12-604
$NIX25   = ******   (X       9-428
$NTLFC     000164 RG         6-119     #6-120    *16-806
$NTLPT   = ******   (X       7-234
$PDVID   = ******   (X       7-247
$PDVNM   = ******   GX       7-246
$PDVTA   = ******   GX       7-250
$PSIPT   = ******   GX       7-241
$PUMR    = ******   GX       7-218
$QASTT   = ******   GX       7-221
$SLTMA   = ******   GX       7-232
$SLTNM   = ******   GX       7-233
$SRSTD   = ******   GX       7-215
$TKPS    = ******   GX       7-223
$TKTCB   = ******   GX       7-220
$TK100   = ******   GX       7-238
$T1SCL   = ******   GX       7-243
$T100C   = ******   GX       7-236
$T100Q   = ******   GX       7-237
$UMRPT   = ******   GX       7-214
$$$ARG   = 000002          #13-689     13-689    #13-689     13-689    #13-689     13-689    #13-689     13-689     13-689
                           #13-689     13-689     13-689     13-689     13-689     13-689     13-689     13-689     13-689
                            13-689     13-689    #13-689     13-689     13-689     13-689     13-689     13-689     13-689
                            13-689     13-689     13-689     13-689    #13-689     13-689     13-689     13-689     13-689
                            13-689     13-689     13-689     13-689     13-689     13-689
$$$GLB   = ******            6-104      6-109      6-110      6-116      6-118      6-119      6-122      6-124
$$$T1    = 000006           #6-104      6-104     #6-110      6-110      6-110      6-110     #6-116      6-116     #6-118
                             6-118      6-118      6-118     #6-119      6-119      6-119      6-119     #6-122      6-122
                             6-122      6-122     #6-124      6-124      6-124      6-124
$$$$     = 000000          #17-846     17-851
.BASEB   = 140000           #5-77
.CXALL     000062   G       #5-72
.CXCSR     000004   G       #5-72
.CXLBR     000060   G       #5-72
.CXLB1     000030   G       #5-72
.CXLB2     000044   G       #5-72
.CXLEN     000064   G       #5-72
.CXPCB     000006   G       #5-72
.CXSYM     000010   G       #5-72
```

```
                                                                                      N 15
NTINIT      CREATED BY  MACRO  ON 5-SEP-85 AT 11:04          PAGE 7
```

```
NTITIM      CREATED BY  MACRO  ON 29-JUN-85 AT 03:28      PAGE 2
MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME        REFERENCES

CALL               6-96         6-114        6-117
CALLR              7-152
CLKDF$            #5-60         5-63
MTPS               6-98         6-111
NHWDF$            #5-60         5-62
RETURN             6-119        7-153        8-194
SWSTK$             6-96
```

```
                                                                    N  1
VUNL        CREATED BY  MACRO  ON 16-SEP-85 AT 07:31       PAGE 1

SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL   VALUE              REFERENCES

AB       = ******  GX        14-793      14-794
A$LEN      000004            10-488
A.ACC      000020           #5-81        21-1520
A.LNK      000000           #5-81
A.NAM      000002           #5-81
A.PUD      000010           #5-81
A.REM      000012           #5-81
A.UCB      000010           #5-81
BLOCK      000012 R         #6-138       22-1594     22-1626     22-1630     22-1639     23-1725     23-1727     23-1736     23-1755
                             23-1759     23-1773
CLQREM   = ******  GX        21-1284
COUNT      000116 R         #6-153      *23-1699    *23-1712    *23-1714    *23-1722
CSRUNL     003440 R          14-686     #20-1204
DDMAC      000122 R         #6-159       14-755     *15-877
DEAECL     001226 R          8-379      #11-523
DEAXPT     000564 R          8-374      #10-441
DEVEC      000000 R         #7-203
DLCAC      000120 R         #6-158       14-772     *15-876
DSIZE    = ******  GX        21-1272
D$ANN      000000            21-1512
D$END    = 000104            10-485      21-1389     21-1509     21-1538
D$NBEA     000056            10-487
D$NBRA     000054            10-486
D$RNN      000002            21-1541
D.DSP    = ******  GX       *21-1482
D.NAM    = ******  GX        21-1471     21-1473     21-1475     21-1477
D.PCB    = ******  GX        21-1469     21-1480    *21-1483
ERRA       001606 R          13-626     #13-628
ERR44      003126 R          17-1065    #17-1068
ERR77      003664 R          20-1234    #20-1236
E$NBR      000014           #5-85
E$NBS      000020           #5-85
E$NCR      000034           #5-85
E$NCS      000036           #5-85
E$NIC      000044           #5-85
E$NLEN     000050           #5-85        11-527
E$NLLA     000012           #5-85
E$NLNK     000000           #5-85
E$NML      000040           #5-85
E$NMR      000024           #5-85
E$NMS      000030           #5-85
E$NNOD     000002           #5-85
E$NRT      000042           #5-85
E$NRTP     000005           #5-85
E$NSEG     000010           #5-85
E$NTIM     000046           #5-85
E$NUSE     000004           #5-85
E$STRT     000006           #5-85
FATAL    = ******  GX        20-1234
FE.CEX   = ******  GX        21-1256     21-1386
FF.ADD   = 000020           #5-80        5-80
FF.DIR   = 000040           #5-80        5-80
```

```
     86                                     .SBTTL  LOCAL SYMBOL DEFINITIONS
     87
     88                                 ;
     89                                 ; SAVED REGISTER OFFSETS FOR $SAVAL
     90                                 ;
     91         000002                  R$R0 = 2
     92         000004                  R$R1 = 4
     93         000006                  R$R2 = 6
     94         000010                  R$R3 = 10
     95         000012                  R$R4 = 12
     96         000014                  R$R5 = 14
```

```
                                        .SBTTL  DEALST - DEALLOCATE LIST OF DATA BLOCKS

                                ;+
                                ;
                                ; DEALST - DEALLOCATE LIST OF DATA BLOCKS
                                ;
                                ; INPUTS:
                                ;       R1 - LENGTH OF BLOCK TO DEALLOCATE
                                ;       R3 - ADDRESS OF LISTHEAD
                                ;
                                ; OUTPUTS:
                                ;       ALL BLOCKS IN THE LIST ARE DEALLOCATED AND THE LISTHEAD IS ZEROED
                                ;       R0,R2 DESTROYED
                                ;
                                ;-
002046                          DEALST: CALL    $SAVAL          ; SAVE ALL REGISTERS
002052                                  SAVMAP
002056                                  GETRV   R3,#2           ; READ IN LISTHEAD
002074  016746  000000G                 MOV     $BFR,-(SP)      ; SAVE FIRST BLOCK ADDRESS
002100  005067  000000G                 CLR     $BFR            ; AND ZERO LISTHEAD
002104                                  PUTRC                   ; WRITE UPDATED LISTHEAD

002110  012600                  10$:    MOV     (SP)+,R0        ; GET THE NEXT BLOCK IN LIST
002112  001422                          BEQ     20$             ; IF EQ, END OF LIST
002114                                  CEACC$  R0,R2           ; CONVERT TO MAPPED ADDRESS
002124                                  GETAD   R2,#2           ; READ IN POINTER TO NEXT BLOCK
002142  016746  000000G                 MOV     $BFR,-(SP)      ; GET THE NEXT BLOCK IN LIST

002146                                  SAVRG   <R1>            ; SAVE LENGTH OF BLOCK
002150                                  CALL    $DEPOL          ; DE-ALLOCATE THE BLOCK
002154                                  RESRG   <R1>            ; RESTORE LENGTH OF BLOCK
002156  000754                          BR      10$             ; AND GO AGAIN

002160                          20$:    RESMAP
002164                          30$:    RETURN

        000001                          .END
```

```
    257
    258                                             ;+
    259                                             ; $TMRDY - MARK SLT AS READY AND ACTIVE IF APPROPRIATE
    260                                             ;
    261                                             ; INPUTS:
    262                                             ;       R0=SLT ADDRESS
    263                                             ;
    264                                             ; OUTPUTS:
    265                                             ;       R1,P2=DESTROYED
    266                                             ;       LF.RDY SET IF DDM, DLC, DDM LINE TABLE, AND DLC LINE TABLE ARE IN CORE
    267                                             ;-
    268 000434                                      $TMRDY::
    269 000434  016701  000000G                             MOV     $DLCPV,R1       ; GET DLC PDV ADDRESS
    270 000440  001403                                      BEQ     5$              ; IF EQ, COMBINATION DDM/DLC
    271 000442  005761  000000G                             TST     Z.PCB(R1)       ; IS DLC CORE-RESIDENT?
    272 000446  001420                                      BEQ     50$             ; IF EQ, NO
    273 000450  016701  000000G                     5$:     MOV     $DDMPV,R1       ; GET DDM PDV ADDRESS
    274 000454  005761  000000G                             TST     Z.PCB(R1)       ; IS DDM CORE-RESIDENT?
    275 000460  001413                                      BEQ     50$             ; IF EQ, NO
    276 000462  005760  000000G                             TST     L.DDS(R0)       ; IS DDM LINE TABLE PRESENT?
    277 000466  001410                                      BEQ     50$             ; IF EQ, NO
    278 000470  005760  000000G                             TST     L.DLS(R0)       ; IS DLC LINE TABLE PRESENT?
    279 000474  001405                                      BEQ     50$             ; IF EQ, NO
    280 000476                                              CALL    MANINI          ; INITIALIZE MANAGEMENT FIELDS
    281 000502  052760  000000G 000000G                     BIS     #LF.RDY,L.FLG(R0) ; MARK LINE AS READY
    282 000510                                      50$:    RETURN
```

```
VXIT       CREATED BY  MACRO  ON 29-JUN-85 AT 03:12     PAGE 2
MACRO CROSS REFERENCE                                  CREF   04.00

MACRO NAME      REFERENCES

CALL             5-107      5-109      5-111      5-113      5-150      5-156      5-163      5-172      5-180      5-197
CALLR            5-87       5-91
CLOSE$          #5-71       5-186
DHBDF$          #5-72       5-75
DIR$            #5-191      5-191
ERBLK$          #5-72       5-200
ERMSG$          #5-72       5-199
ERRPT$          #5-72       5-197
ERR$            #5-186      5-186
EXIT$S          #5-71       5-191
GETRC           #5-71       5-172
GETRV           #5-71       5-150
LDFDB$          #5-186      5-186
PUTAD           #5-71
PUTRC           #5-71       5-107      5-109      5-111      5-113
RAND            #5-107      5-107     #5-109      5-109     #5-111      5-111     #5-113      5-113     #5-150      5-150
                #5-172      5-172
SWTCHI          #5-71       5-145
SWTCHO          #5-71       5-106      5-108      5-110      5-112      5-146
```

```
    112 000216   000411                          BR      40$
    113 000220   016767   000060G 000010G 30$:   MOV     $BFR+60,P.SUB+$BFR        ; UNLINK OUR SUB-PCB
    114 000226                                   PUTRC                             ; REWRITE UPDATED PCB
    115 000232   016701   000000G                MOV     .PLGTH,R1                 ; GET PCB LENGTH
    116 000236                                   CALL    $DEA16                    ; DE-ALLOCATE SUB-PCB
    117 000242                            40$:   RETURN
    118
    119          000001                          .END
```

```
                ;+
                ; *** - INILIN - REINITIALIZE HRC
                ;
                ; THIS SUBROUTINE IS USED AS A GENERALIZED INTERFACE TO REQUEST HRC...
                ; TO REINITIALIZE ITSELF AFTER THE I/O DATA BASE HAS BEEN CHANGED.
                ;
                ; INPUT:
                ;       NUNE.
                ;
                ; OUTPUT:
                ;       HRC... INITIALIZED OR ERROR MESSAGE PRINTED
                ;
                ;-

                        .IF DF  R$$MPL
                INILIN: CALL    $SAVAL                  ; SAVE REG'S
                        MOV     @SLTMA,R5               ; GET START OF SLT'S
                        MOV     @SLTNM,R4               ; GET NUMBER OF SLT'S

                1$:     MOV     (R5)+,R3                ; GET NEXT SLT ADDRESS
                        BIT     #LF.RDY,L.FLG(R3)       ; IS AT LEAST ONE LINE LOADED ?
                        BNE     5$                      ; IF YES - INIT HRC...
                        DEC     R4                      ; COUNT ONE LESS SLT
                        BNE     1$                      ; IF MORE - BRANCH
                        BR      20$                     ; CONTINUE

                5$:     ALUN$S  #3,#"RD,#0              ; ASSIGN LUN TO RDO:
                        BCS     20$                     ; IF CS, EXIT

                        QIOW$S  #IO.ATT,#3,#32.,,#STATUS  ; ATTACH TO THE RDO:
                        BCS     20$                     ; IF CS, ERROR
                        QIOW$S  #IO.HRC,#3,#32.,,#STATUS,,<#M$INIT,#M$INIT>
                                                        ; REQUEST HRC... TO
                                                        ; REINITIALIZE ITSELF
                        ROR     R1                      ; SAVE ERROR STATUS
                        QIOW$S  #IO.DET,#3,#32.         ; DETACH FROM RECONFIGURATION DRIVER
                        ROL     R1                      ; RESTORE CARRY
                        BCS     10$                     ; IF CS, EXIT
                        TSTB    STATUS                  ; WAS THE QIO SUCCESSFUL ?
                        BPL     20$                     ; IF PL, YES
                10$:    EMSG$R  11                      ; ELSE, INDICATE ERROR

                20$:    RETURN
                        .ENDC


        000001          .END
```

```
NTICSR  MACRO V05.03b Saturday 29-Jun-85 03:25  Page 6-2
Symbol table

A$$CHK= 000000         LF.ENA= 002000       L.DDM   000002       PR6   = 000300       ZF.DDM= 000001
A$$CPS= 000000         LF.LPB= 001000       L.DDS   000004       PR7   = 000340       ZF.DIA= 004000
A$$PRI= 000000         LF.MDC= 000100       L.DLC   000003       PS    = 177776       ZF.DLC= 000002
A$$TRP= 000000         LF.MFL= 004000       L.DLM   000006       P$$P45= 000000       ZF.DVP= 100000
BUFUMP= 172354         LF.MTP= 000020       L.DLS   000010       P$$WRD= 000000       ZF.INI= 040000
CMODE = 140000         LF.PAC= 000200       L.FLG   000000       Q$$OPT= 000010       ZF.KMX= 000020
CSRDIS  000006R    002 LF.RDY= 040000       L.KRBA  000016       R$$DER= 000000       ZF.LLC= 000004
CSRENA  000000R    002 LF.REA= 010000       L.LEN = 000022       R$$K11= 000001       ZF.LMC= 000100
CSRSLT  000016R    002 LF.SER= 000040       L.MPF   000022       R$$SND= 000000       ZF.MAN= 020000
CSRTAB  000014R    002 LF.TIM= 000010       L.NMST  000020       R$$11M= 000000       ZF.MFL= 000010
CSRTRP  000274R        LF.UNL= 020000       L.NSTA  000014       SF.ACT= 000200       ZF.MTM= 000400
C$$CKP= 000000         LF.X2P= 000000       L.OWNR  000021       SF.ENA= 000100       ZF.MUX= 000040
C$$ORE= 000400         LN.CLO= 000000       L.UNT   000013       SF.LPB= 000004       ZF.PSE= 002000
C$$RSH= 177564         LN.DUM= 000005       MPAR  = 172100       SF.MFL= 000040       ZF.SLI= 010000
D$$BUG= 177514         LN.LOA= 000004       MPCSR = 177746       SF.PAC= 000020       ZF.TIM= 000200
D$$ISK= 000000         LN.LOO= 000003       M$$CRB= 000124       SF.REA= 000010       ZF.X3P= 000000
D$$L11= 000001         LN.OAU= 000003       M$$CRX= 000000       SF.SER= 000001       ZS.ASN= 100000
D$$YNC= 000000         LN.OFF= 000001       M$$FCS= 000000       SF.SVC= 000002       ZS.BSY= 140000
D$$YNM= 000000         LN.ON = 000000       M$$MGE= 000000       SF.UNL= 000040       Z.AVL   000014
E$$XPR= 000000         LN.OOP= 000004       M$$NET= 000000       SWR   = 177570       Z.DAT   000016
F$$LVL= 000001         LN.OPE= 000001       M$$OVR= 000000       S$$WRG= 000000       Z.DSP   000000
G$$TPP= 000000         LN.REF= 000002       N$$ACC= 000001       S$$YSZ= 007600       Z.FLG   000010
G$$TSS= 000000         LN.SER= 000002       N$$BUF= 000001       S.COST  000001       Z.LEN = 000016
G$$TTK= 000000         LN.STA= 000017       N$$LDV= 000001       S.FLG   000000       Z.LLN   000006
G$$WRD= 000000         LN.SUB= 000360       N$$MCP= 000001       S.LEN   000004       Z.MAP   000020
I$$RAR= 000000         LN.TRI= 000006       N$$MLL= 000001       S.NMST  000002       Z.NAM   000004
I$$RDN= 000000         LR.CTL= ****** GX    N$$MOV= 000010       S.OWNR  000003       Z.PCB   000012
KISAR0= 172340         LR.LIN= ****** GX    N$$NCT= 000001       S.VTTA= 000002       Z.SCH   000007
KISAR6= 172354         LR.STS= ****** GX    N$$PEM= 000001       S.VTTL= 000004       $ERR20  000020RG      002
KSAR6 = ****** GX      LR.UNT= ****** GX    PDVTA = ****** GX    TPS   = 177564       $NERDN= ****** GX
K$$CNT= 177546         LX.LIN= ****** GX    PIRQ  = 177772       T$$KMG= 000000       $NICSR  000000RG
K$$CSR= 177546         L$$ASG= 000000       PMODE = 030000       T$$MIN= 000000       $NIERA= ****** GX
K$$LDC= 000000         L$$DRV= 000000       PR0   = 000000       UBMPR = 170200       $NIERR= ****** GX
K$$TPS= 000074         L$$P11= 000001       PR1   = 000040       UISAR0= 177640       $NINTL= ****** GX
LD$LP = 000000         L$$11R= 000000       PR2   = 000100       UISAR1= 177642       $NTLFC= ****** GX
LF.ACT= 100000         L.COST  000015       PR3   = 000140       V$$CTR= 001000       $$$OST= 000006
LF.BRO= 000400         L.CTL   000012       PR4   = 000200       X$$DBT= 000000       .BASEB= 140000
LF.BWT= 000007         L.CVA   177776       PR5   = 000240       ZF.COU= 001000

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000302    001   (RW,I,LCL,REL,CON)
DATA    000052    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 14216 Words  ( 56 Pages)
Size of core pool: 15496 Words  ( 59 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:12.48
SY:NTICSR.V2,[132,134]NTICSR/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTICSR
```

```
000364  006316                           ASL     (SP)               ; CONVERT VECTOR TABLE TO BYTES
000366  062600                           ADD     (SP)+,R0           ; POINT PAST PLD VECTOR
000370  005020                   140$:   CLR     (R0)+              ; ZERO OUT THE PLD'S
000372                                   SOB     R1,140$            ; LOOP R1 TIMES


                                         .ENDC   ; NDF R$$11S


000376  017701  000000G          150$:   MOV     @DECPT,R1          ; GET DECNET HOMEBLOCK ADDRESS
000402  005761  000054                   TST     D$NBRA(R1)         ; ANY ROUTER ADJACECIES
000406  001426                           BEQ     270$               ; IF NOT - BRANCH
000410  016101  000054                   MOV     D$NBRA(R1),R1      ; GET NUMBER OF BROADCAST ROUTER ADJACENCIES
000414  062701  000077                   ADD     #77,R1             ; ROUND UP TO 32. WORD BLOCK
000420  042701  000077                   BIC     #77,R1             ; ...
000424  005000                           CLR     R0                 ; ASSUME FAILURE
000426                                   SWSTK$  260$               ; MAP TO SYSTEM STATE
000432                                   CALL    @ALOCB             ;* ALLOCATE SPACE FROM DSR FOR P
000436  103406                           BCS     250$               ;* IF FAILURE - BRANCH
000440  010066  000002                   MOV     R0,2(SP)           ;* INDICATE SUCCESS THRU USER R0
000444  006201                           ASR     R1                 ;* GET THE NUMBER OF WORDS ALLOCATED
000446  005020                   240$:   CLR     (R0)+              ;* CLEAR PRIORITY DATA BASE
000450                                   SOB     R1,240$            ;* ALL OF IT
000454                           250$:   RETURN                     ;* LEAVE SYSTEM STATE

000456  010062  000076           260$:   MOV     R0,N$PRI(R2)       ; SET UP XPT DATA DASE WITH PRIORITY ADDRESS
000462  001447                           BEQ     370$               ; IF ALLOCATION FAILURE - REPORT ERROR

000464                           270$:   DIR$    #LONACP            ;REQUEST THE ACP
000472  103003                           BCC     290$               ; IF SUCCESS - BRANCH
000474  012703  000566'                  MOV     #350$,R3           ; ELSE - SET UP ERROR POINTER
000500  000416                           BR      320$               ; CONTINUE

000502  016702  000000G          290$:   MOV     $DBECL,R2          ; GET ADDRESS OF ECL'S DDB

000506  132762  000040  000005   300$:   BITB    #NF$MOU,N$FLG(R2)  ; HAS THE NETWORK BEEN MOUNTED YET?
000514  001004                           BNE     310$               ; BR IF YES
000516                                   WSIG$S                     ; WAIT FOR NETACP TO RUN
000524  000770                           BR      300$               ; ...
000526                           310$:   CALL    $NIMFE             ; DO MARKED-FOR-ENABLE SCAN
000532  000241                           CLC                        ; ..SUCCESS - ALL DONE
000534                                   RETURN


                                 ;
                                 ; DE-ALLOCATE THE PLD DATA BASE
                                 ;
000536                           320$:   .IF NDF R$$11S

000536                                   SWSTK$  330$               ;* ENTER KERNEL MODE
000542  016700  000000'                  MOV     PLDADR,R0          ;* GET THE PLD DB ADDRESS
000546  016701  000002'                  MOV     PLDLEN,R1          ;* AND LENGTH
000552                                   CALLR   @DEACB             ;* DE-ALLOCATE BLOCK

                                         .ENDC   ; NDF R$$11S

000556  000113                   330$:   JMP     (R3)               ; DE-ALLOCATE THE VCB OR PRINT ERMSG
                                 ;
000560                           340$:   EMSG$R  15                 ;ALUN$ FAILURE OR NS LOAD FAILURE
```

```
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:08
Table of contents
```

```
     5-   54    DATA
     6-   98    MARKED FOR ENABLE SCAN
     7-  255    ENABLE A LINE
     8-  310    DLMSET - SET UP COUNTER BLOCK FIELDS FOR DLM CIRCUITS
```

N 11

```
NTIMFE     CREATED BY  MACRO  ON 24-JUL-85 AT 15:09       PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE           REFERENCES

LF.TIM  = 000010        #5-74
LF.UNL  = 020000        #5-74
LF.X2P  = 000000        #5-74
LLCRS   = ******  GX     7-297
LLCTA   = ******  GX     6-121
LN.CLO  = 000000        #5-74
LN.DUM  = 000005        #5-74
LN.LOA  = 000004        #5-74
LN.LOO  = 000003        #5-74
LN.OAU  = 000003        #5-74
LN.OFF  = 000001        #5-74
LN.ON   = 000000        #5-74       6-202      6-244
LN.OOP  = 000004        #5-74
LN.OPE  = 000001        #5-74
LN.REF  = 000002        #5-74
LN.SER  = 000002        #5-74
LN.STA  = 000017        #5-74
LN.SUB  = 000360        #5-74
LN.TRI  = 000006        #5-74
L.COST    000015        #5-74
L.CTL     000012        #5-74       6-166
L.CVA     177776        #5-74
L.DDM     000002        #5-74
L DDS     000004        #5-74
L.DLC     000003        #5-74       6-129
L.DLM     000006        #5-74
L.DLS     000010        #5-74
L.FLG     000000        #5-74       6-125      6-132      6-135     *6-210      6-214     *6-236     *6-243
L.KRBA    000016        #5-74
L.LEN   = 000022        #5-74
L.MPF     000022        #5-74       6-137      6-168
L.NMST    000020        #5-74      *6-244
L.NSTA    000014        #5-74       6-158
L.OWNR    000021        #5-74
L.UNT     000013        #5-74
M$$EXT  = 000000        #5-60
M$$MGE  = 000000         5-60       8-347
NF$MOU  = 000040         6-183      6-227
N$FLG     000005         6-183      6-227
N$TLC     000100         8-344      8-345      8-346
N$$VCT  = ******         7-299      8-333      8-342      8-344      8-359      8-371      8-374
PDVTA   = ******  GX     6-130      6-177      6-221      7-275
PSIPT   = ******  GX     8-331
PX$DLM  = 000200         8-355
PX$SVC  = 000100         8-358
R$$MPL  = ******         6-143
R$$11D  = ******         5-77
R$$11M  = 000000         5-77       8-347
R$$11S  = ******         5-77       7-279      8-327
SF.ACT  = 000200        #5-74       6-137      6-201
SF.ENA  = 000100        #5-74       6-169      6-192
SF.LPB  = 000004        #5-74
```

```
NTIMFL  MACRO V05.03b Saturday 29-Jun-85 03:27  Page 9-2
Symbol table

Size of work file: 15863 Words  ( 62 Pages)
Size of core pool: 16552 Words  ( 63 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:14.11
SY:NTIMFL.V2,[132,134]NTIMFL/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTIMFL
```

```
    213
    214 000614  000000G                 UMRPT:: $UMRPT
    215 000616  000000G                 SRSTD:: $SRSTD
    216 000620  000000G                 ALOCB:: $ALOCB
    217 000622  000000G                 DEACB:: $DEACB
    218 000624  000000G                 PUMR::  $PUMR
    219 000626  000000G                 HEADR:: $HEADR
    220 000630  000000G                 TKTCB:: $TKTCB
    221 000632  000000G                 QASTT:: $QASTT
    222 000634  000000G                 CLKHD:: $CLKHD
    223 000636  000000G                 TKPS::  $TKPS
    224 000640  000000G                 ABTIM:: $ABTIM
    225 000642  000000G                 DEVHD:: $DEVHD
    226 000644  000000G                 LDPWF:: $LDPWF
    227
    228                                         .IF DF  R$$MPL
    229                                 MPPRO:: $MPPRO
    230                                         .ENDC
    231
    232 000646  000000G                 SLTMA:: $SLTMA                  ; COMM EXEC VECTORED VALUES
    233 000650  000000G                 SLTNM:: $SLTNM
    234 000652  000000G                 NTLPT:: $NTLPT
    235 000654  000000G                 CALLX:: $CALLX
    236 000656  000000G                 T100C:: $T100C
    237 000660  000000G                 T100Q:: $T100Q
    238 000662  000000G                 TK100:: $TK100
    239 000664  000000G                 CCBGT:: $CCBGT
    240 000666  000000G                 LLCRS:: $LLCRS
    241 000670  000000G                 PSIPT:: $PSIPT
    242 000672  000000G                 CEACC:: $CEACC
    243 000674  000000G                 T1SCL:: $T1SCL
    244 000676  000000G                 LGSTT:: $LGSTT
    245 000700  000000G                 DECPT:: $DECPT
    246 000702  000000G                 PDVNM:: $PDVNM
    247 000704  000000G                 PDVID:: $PDVID
    248 000706  000000G                 LLCAL:: $LLCAL
    249 000710  000000G                 LLCTA:: $LLCTA
    250 000712  000000G                 PDVTA:: $PDVTA
    251
    252         000043                          VECLN == <<.-EXEVEC>/2>
    253
    254                                 ;
    255                                 ; START OF PURE CODE
    256                                 ;
    257 000000                                  .PSECT
```

```
                                        .SBTTL  RCV DATA AST

                                ;+
                                ; NTIRDA - NTINIT RCV DATA AST
                                ;
                                ; INPUTS:
                                ;       NONE
                                ;
                                ; OUTPUTS:
                                ;       THE RCV DATA SUBROUTINE IS CALLED UNTIL THE RCV DATA QUEUE
                                ;         IS EMPTIED.
                                ;-

                                        .IF NDF R$$11S

 001042  010046                 NTIRDA: MOV     R0,-(SP)                ;SAVE R0
 001044                         10$:    DIR$    #NTIRCV                 ;RECEIVE DATA
 001052  103403                         BCS     20$                     ; .. NONE
 001054                                 CALL    NTIRDT                  ;PROCESS THE DATA
 001060  000771                         BR      10$                     ;TRY FOR SOME MORE
 001062  012600                 20$:    MOV     (SP)+,R0                ;RESTORE R0
 001064                                 ASTX$S                          ;EXIT FROM AST

                                        .ENDC
```

N 15

```
NTINIT     CREATED BY  MACRO  ON 5-SEP-85 AT 11:04        PAGE 7
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

.CXUNL     000024   G       #5-72
.CXVEC     000026   G       #5-72
.CXVSM     000016   G       #5-72
.NTGCD     000000           #5-72
.NTPCB     000000   G       #5-72
.NTUMR     000002   G       #5-72
.SBPCB     000066   G       #5-72
```

```
B  1
C  1
D  1
E  1
F  1
G  1
H  1
I  1
J  1
K  1
L  1
M  1
N  1
B  2
C  2
D  2
E  2
F  2
G  2
H  2
I  2
J  2   VUNPSI
K  2
L  2
M  2
N  2
B  3
C  3
D  3
E  3
F  3
G  3
H  3
I  3
J  3
K  3
L  3
M  3
N  3
B  4
C  4
D  4
E  4
F  4
G  4   VVCT
H  4
I  4
J  4
K  4
L  4
M  4
N  4
B  5
C  5
D  5
E  5
F  5
G  5   VXIT
H  5
I  5
```

```
J  5
K  5
L  5
M  5
N  5
B  6   VXLB
C  6
D  6
E  6
F  6
G  6
H  6
I  6
J  6
K  6   VXPR
L  6
M  6
N  6
B  7
C  7
D  7
E  7
F  7   NTICEX
G  7
H  7
I  7
J  7
K  7
L  7
M  7
N  7
B  8
C  8
D  8
E  8
F  8
G  8
H  8   NTICSR
I  8
J  8
K  8
L  8
M  8
N  8
B  9
C  9
D  9
E  9
F  9   NTILON
G  9
H  9
I  9
J  9
K  9
L  9
M  9
N  9
B 10
C 10
D 10
```

```
E 10
F 10
G 10
H 10
I 10
J 10
K 10
L 10
M 10   NTIMFE
N 10
B 11
C 11
D 11
E 11
F 11
G 11
H 11
I 11
J 11
K 11
L 11
M 11
N 11
B 12
C 12
D 12
E 12   NTIMFL
F 12
G 12
H 12
I 12
J 12
K 12
L 12
M 12
N 12
B 13
C 13
D 13
E 13
F 13
G 13   NTINIT
H 13
I 13
J 13
K 13
L 13
M 13
N 13
B 14
C 14
D 14
E 14
F 14
G 14
H 14
I 14
J 14
K 14
L 14
```

```
M 14
N 14
B 15
C 15
D 15
E 15
F 15
G 15
H 15
I 15
J 15
K 15
L 15
M 15
N 15
B 16
C 16   NTITIM
D 16
E 16
F 16
G 16
H 16
I 16
J 16
K 16
L 16
M 16
```

```
**FILE**ID**NTIUMR




NN      NN   TTTTTTTTTT     IIIIII     UU      UU   MM      MM   RRRRRRRR
NN      NN   TTTTTTTTTT     IIIIII     UU      UU   MM      MM   RRRRRRRR
NN      NN       TT           II       UU      UU   MMMM  MMMM   RR      RR
NN      NN       TT           II       UU      UU   MMMM  MMMM   RR      RR
NNNN    NN       TT           II       UU      UU   MM  MM  MM   RR      RR
NNNN    NN       TT           II       UU      UU   MM  MM  MM   RR      RR
NN  NN  NN       TT           II       UU      UU   MM      MM   RRRRRRRR
NN  NN  NN       TT           II       UU      UU   MM      MM   RRRRRRRR
NN    NNNN       TT           II       UU      UU   MM      MM   RR  RR
NN    NNNN       TT           II       UU      UU   MM      MM   RR  RR
NN      NN       TT           II       UU      UU   MM      MM   RR    RR         ....
NN      NN       TT           II       UU      UU   MM      MM   RR    RR         ....
NN      NN       TT         IIIIII     UUUUUUUUUU   MM      MM   RR      RR       ....
NN      NN       TT         IIIIII     UUUUUUUUUU   MM      MM   RR      RR       ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
NTIUMR  MACRO V05.03b Saturday 29-Jun-85 03:28
Table of contents
```

```
                                      .SBTTL  LOCAL DATA
                              ;
                              ; LOCAL DATA (IMPURE)
                              ;

                              ;
                              ; CONSTANT DEFINTIONS
                              ;
             000002           R$R0 = 2                        ; OFFSET OF SAVED R0
             000004           R$R1 = 4                        ; OFFSET OF SAVED R1
             000006           R$R2 = 6                        ; OFFSET OF SAVED R2
             000010           R$R3 = 10                       ; OFFSET OF SAVED R3
             000012           R$R4 = 12                       ; OFFSET OF SAVED R4
             000014           R$R5 = 14                       ; OFFSET OF SAVED R5

    000000                            .PSECT  DATA,D

                              ;
                              ; DIRECTIVES
                              ;
    000000                    LONACP: RUN$    X25ACP,,,,,1,1          ;DIRECTIVE TO REQUEST X25ACP
    000026                    ASSIGN: ALUN$   2,NW,0                  ;ASSIGN LUN 2 TO NW:

                              ;
                              ; ERROR MESSAGES
                              ;
                                      .ENABL  LC
    000036                    NTLER$  ,40,NONE,$NIERR,,,<NW: Assignment Failure>
    000076                    NTLER$  ,41,NONE,$NIERR,,,<VCB Allocation Failure>
    000136                    NTLER$  ,42,NONE,$NIERR,,,<X25 ACP (X25ACP) Not Installed>
                                      .DSABL  LC

                              ;
                              ; START OF PURE CODE
                              ;
    000000                            .PSECT
```

```
                        .SBTTL  SET CEX FUNCTION

                ;+
                ; $NICEX - SET CEX FUNCTION ENTRY POINT
                ;
                ; THE SAME ENTRY POINT GLOBAL IS DEFINED FOR BOTH RSX11M AND RSX11S
                ; SYSTEMS.
                ;
                ; INPUTS:
                ;       ON AN RSX11M SYSTEM ONLY, FE.CEX IN "$FMASK" INDICATES
                ;         WHETHER CEX IS ALREADY LOADED (1) OR NOT (0).
                ;       ON AN RSX11S SYSTEM, CEX IS ASSUMED TO BE ALREADY
                ;         LOADED.
                ;       LS.CXO IN "$NIFNC" INDICATES WHETHER THE FUNCTION IS
                ;         LOAD CEX ONLY (0) OR ALSO PERFORM LOCAL ON (1).
                ;
                ; OUTPUTS:
                ;       CEX IS LOADED, UMR'S ARE LOADED, CSR'S ARE CHECKED
                ;         AND THE TIMERS ARE STARTED, AS NECESSARY.
                ;       ON RSX11M SYSTEMS WHERE CEX IS NOT YET LOADED, NTL IS
                ;         CALLED TO PERFORM INITIALIZATION, AND A "MARKED FOR
                ;         LOAD" AND "MARKED FOR ENABLE" SCAN IS PERFORMED.
                ;       ON RSX11M SYSTEMS WHERE CEX IS ALREADY LOADED, LOCAL
                ;         INITIALIZATION IS PERFORMED.
                ;       ON AN RSX11S SYSTEM, LOCAL INITIALIZATION ONLY IS
                ;         PERFORMED (FALL THROUGH TO LOCAL INIT CODE).
                ;       CARRY IS SET IF LOAD OR INITIALIZATION FAILED.
                ;-

000000          $NICEX::.IF     NDF,R$$11S              ;*-THE FOLLOWING IS RSX11M CODE-*

                        BIT     #FE.CEX,@FMASK          ;IS CEX LOADED ALREADY?
                        BNE     NTILIN                  ;BR IF YES, PERFORM LOCAL INIT
                        MOV     #LS.CEX+LS.NTI,$NTLFC   ;SET NTL FUNCTION CODE
                        CALL    $NINTL                  ;SEND DATA TO NTL
                        BCS     20$                     ;BR ON ERROR (NO NTL)
                        TST     $NTLFC                  ;WHAT WAS NTL'S RESPONSE
                        SEC                             ; ASSUME FAILURE
                        BMI     20$                     ;BR IF BAD RESPONSE
                        BIT     #LS.TOP,$NTLFC          ; COPY TOP-LOADING FLAG
                        BEQ     10$                     ; FROM NTL'S ANSWER
                        INCB    $NIFLG+1                ; TO HIGH BYTE OF $NIFLG
                10$:    CALL    FNDPDV                  ;FIND NSP PROCESSES (ECL & XPT)
                        CALL    $NIMFL                  ;PERFORM MARKED FOR LOAD SCAN
                        BCS     30$                     ;BR IF LOAD FAIL, REMOVE CEX
                        CALL    $NITIM                  ;START THE TIMERS IF NECESSARY
                20$:    RETURN                          ;RETURN TO MAIN LINE CODE

                30$:    MOV     #LX.CEX,$NTLFC          ;IF LOAD ERROR, REMOVE CEX
                        CALL    $NINTL                  ;LET NTL DO IT
                        EMSG$R  10                      ;TELL USER AND RETURN W/ CARRY SET

                        .ENDC                           ;*-END OF RSX11M CODE-*
```

```
             .TITLE  NTICSR
             .IDENT  /V05.00/

        ;
        ; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
        ;
        ;
        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
        ; TRANSFERRED.
        ;
        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
        ; CORPORATION.
        ;
        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
        ;
        ; MODULE DESCRIPTION:
        ;
        ;       NTINIT - SCAN CSR LIST AND TEST DEVICES FOR ONLINE
        ;
        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
        ;
        ; IDENT HISTORY:
        ;
        ; 1.00  27-FEB-78
        ;       VERSION 2.0 RELEASE
        ;
        ; 2.00  14-DEC-79
        ;       DECNET-11M/S V3.0
        ;       DECNET-11M-PLUS V1.0
        ;
        ; 3.00  16-APR-82
        ;       DECNET-11M V3.1
        ;       DECNET-11M-PLUS V1.1
        ;
        ; 4.00  07-NOV-83
        ;       DECNET-11M V4.0
        ;       DECNET-11M-PLUS V2.0
        ;
        ; 5.00  22-JUL-85
        ;       DECnet-11M/S V4.2
        ;       DECnet-11M-Plus V3.0
        ;       DECnet-Micro/RSX V1.0
        ;
        ;

```

```
                                                .SBTTL  NODE INFO INITIALIZATION

                                        ;+
                                        ; $NIDLI - DOWN-LINE LOAD INITIALIZATION
                                        ;
                                        ; PLEASE NOTE THAT THIS CODE IS ALWAYS ASSEMBLED FOR AN RSX-11S SYSTEM.
                                        ; IT SHOULD NEVER BE CALLED (OR INCLUDED) ON AN RSX-11M SYSTEM.
                                        ;
                                        ; INPUTS:
                                        ;       IF "$SYSIZ+3,$SYSIZ+4" CONTAINS A -1, AND LOCATION 000002
                                        ;         CONTAINS A VALUE OTHER THAN 340, THEN THE VALUE CAN BE USED
                                        ;         TO MAP TO MEMORY WHICH CONTAINS THE FOLLOWING.
                                        ;
                                        ;               .BLKW   3.      ;THE NODE NUMBER (OR ZERO)
                                        ;               .BLKW   8.      ;THE NODE NAME (OR ZERO)
                                        ;               .BLKW   8.      ;THE HOST NAME (OR ZERO)
                                        ;               .BLKW   3.      ;THE HOST NODE ADDRESS (OR ZERO)
                                        ;               .BLKW   5.      ;THE TIME
                                        ;
                                        ; OUTPUTS:
                                        ;       THE NODE NUMBER IS PLACED IN D$LNUM OF THE DECNET HOME BLOCK
                                        ;       THE NODE NAME IS PLACED IN D$LNAM OF THE DECNET HOME BLOCK
                                        ;       THE HOST NODE ADDRESS IS PLACED IN D$HOST OF THE DECNET HOME BLOCK
                                        ;       R0,R1 = DESTROYED
                                        ;-

000000  012700  000004G                 $NIDLI::MOV     #$SYSIZ+4,R0            ; POINT AT SYSTEM IMAGE'S LBN
000004  022710  177777                          CMP     #-1,(R0)                ; WERE WE DOWN-LINE SYSTEM LOADED ?
000010  001114                                  BNE     30$                     ; NO ..
000012  121040                                  CMPB    (R0),-(R0)              ; MAYBE ?
000014  001112                                  BNE     30$                     ; NO ..
000016  106010                                  RORB    (R0)                    ; MAKE SURE WE DON'T DO THIS TWICE

                                        ;
                                        ; GET INFORMATION ON A MAPPED SYSTEM
                                        ;
000020  022737  000340  000002                  CMP     #340,@#2                ;IS INFORMATION VALID?
000026  001505                                  BEQ     30$                     ;BR IF NO

000030                                          SWSTK$  30$                     ;;GO TO SYSTEM STATE
000034  013767  000002  172354'                 MOV     @#2,KISAR6              ;;SET BIAS OF INFORMATION IN APR
000042  012701  140000                          MOV     #.BASEB,R1              ;;POINT AT DOWN-LINE LOAD NODE INFO
000046  012737  000340  000002                  MOV     #340,@#2                ;;RESET THE LOCATION

000054  012100                                  MOV     (R1)+,R0                ;;IS THERE A NODE NUMBER PRESENT?
000056  001416                                  BEQ     10$                     ;;IF EQ, NO

000060  016703  000000G                         MOV     $DECPT,R3               ;;GET DECNET HOME BLOCK
000064  010046                                  MOV     R0,-(SP)                ;;SAVE NODE ADDRESS
000066  042716  001777                          BIC     #^CAREA,(SP)            ;;ISOLATE AREA
000072  005726                                  TST     (SP)+                   ;;AREA 0 ?
000074  001005                                  BNE     5$                      ;;IF NO - BRANCH
000076  016346  000014                          MOV     D$LNUM(R3),-(SP)        ;;GET VNP'D NODE ADDRESS
000102  042716  001777                          BIC     #^CAREA,(SP)            ;;ISOLATE AREA
000106  052600                                  BIS     (SP)+,R0                ;;SET AREA
000110  010063  000014                  5$:     MOV     R0,D$LNUM(R3)           ;;SET ADDRESS IN HOME BLOCK
```

```
                                          ASL     (SP)                    ; CONVERT VECTOR TABLE TO BYTES
                                          ADD     (SP)+,R0                ; POINT PAST PLD VECTOR
                                  140$:   CLR     (R0)+                   ; ZERO OUT THE PLD'S
                                          SOB     R1,140$                 ; LOOP R1 TIMES

                                          .ENDC   ; NDF R$$11S



000212  017701  000000G           150$:   MOV     @DECPT,R1               ; GET DECNET HOMEBLOCK ADDRESS
000216  005761  000054                    TST     D$NBRA(R1)              ; ANY ROUTER ADJACECIES
000222  001426                            BEQ     270$                    ; IF NOT - BRANCH
000224  016101  000054                    MOV     D$NBRA(R1),R1           ; GET NUMBER OF BROADCAST ROUTER ADJACENCIES
000230  062701  000077                    ADD     #77,R1                  ; ROUND UP TO 32. WORD BLOCK
000234  042701  000077                    BIC     #77,R1                  ; ...
000240  005000                            CLR     R0                      ; ASSUME FAILURE
000242                                    SWSTK$  260$                    ; MAP TO SYSTEM STATE
000246                                    CALL    @ALOCB                  ;* ALLOCATE SPACE FROM DSR FOR P
000252  103406                            BCS     250$                    ;* IF FAILURE - BRANCH
000254  010066  000002                    MOV     R0,2(SP)                ;* INDICATE SUCCESS THRU USER R0
000260  006201                            ASR     R1                      ;* GET THE NUMBER OF WORDS ALLOCATED
000262  005020                    240$:   CLR     (R0)+                   ;* CLEAR PRIORITY DATA BASE
000264                                    SOB     R1,240$                 ;* ALL OF IT
000270                            250$:   RETURN                          ;* LEAVE SYSTEM STATE

000272  010062  000076            260$:   MOV     R0,N$PRI(R2)            ; SET UP XPT DATA DASE WITH PRIORITY ADDRESS
000276  001437                            BEQ     370$                    ; IF ALLOCATION FAILURE - REPORT ERROR

000300                            270$:   DIR$    #LONACP                 ;REQUEST THE ACP
000306  103003                            BCC     290$                    ; IF SUCCESS - BRANCH
000310  012703  000362'                   MOV     #350$,R3                ; ELSE - SET UP ERROR POINTER
000314  000416                            BR      320$                    ; CONTINUE

000316  016702  000000G           290$:   MOV     $DBECL,R2               ; GET ADDRESS OF ECL'S DDB

000322  132762  000040  000005    300$:   BITB    #NF$MOU,N$FLG(R2)       ; HAS THE NETWORK BEEN MOUNTED YET?
000330  001004                            BNE     310$                    ; BR IF YES
000332                                    WSIG$S                          ; WAIT FOR NETACP TO RUN
000340  000770                            BR      300$                    ; ...
000342                            310$:   CALL    $NIMFE                  ; DO MARKED-FOR-ENABLE SCAN
000346  000241                            CLC                             ; ..SUCCESS - ALL DONE
000350                                    RETURN

                                  ;
                                  ; DE-ALLOCATE THE PLD DATA BASE
                                  ;
000352                            320$:   .IF NDF R$$11S

                                          SWSTK$  330$                    ;* ENTER KERNEL MODE
                                          MOV     PLDADR,R0               ;* GET THE PLD DB ADDRESS
                                          MOV     PLDLEN,R1               ;* AND LENGTH
                                          CALLR   @DEACB                  ;* DE-ALLOCATE BLOCK

                                          .ENDC   ; NDF R$$11S

000352  000113                    330$:   JMP     (R3)                    ; DE-ALLOCATE THE VCB OR PRINT ERMSG
                                  ;
000354                            340$:   EMSG$R  15                      ;ALUN$ FAILURE OR NS LOAD FAILURE
```

```
        .TITLE  NTIMFE - DO MARKED-FOR-ENABLE SCAN
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       NTINIT - MARKED-FOR-ENABLE SCAN
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                                                                   B  8
NTIMFE11S  CREATED BY  MACRO  ON 24-JUL-85 AT 15:52     PAGE 3

SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE            REFERENCES

STANUM    000002 R       #6-91      *7-167     *7-205
STRT    = 000004         #6-84       8-295
S$$BAS  = ******          7-112
S.COST    000001         #6-74
S.FLG     000000         #6-74      *7-192
S.LEN     000004         #6-74
S.NMST    000002         #6-74      *7-202
S.OWNR    000003         #6-74
TMEFN   = 000001         #6-85       7-113      7-114
T$FLAG    000044         #6-72
T$LIF     000013         #6-72
T$LIFL    000013         #6-72
T$LIFO    000013         #6-72
T$LIFS    000013         #6-72
T$LIN     000000         #6-72
T$LIPS    000006         #6-72
T$LLD     000012         #6-72
T$LLDC    000045         #6-72
T$LLDL    000012         #6-72
T$LLDO    000012         #6-72
T$LLDS    000012         #6-72
T$LLEN    000046         #6-72
T$LOPR    000002         #6-72
T$LTCL    000024         #6-72
T$LTIM    000026         #6-72
T$LTPR    000014         #6-72
T$LTPS    000020         #6-72
T$NAPL    000004         #6-72
T$NFE     000000         #6-72
T$NLEN    000010         #6-72
T$NNUL    000002         #6-72
T$NOPL    000006         #6-72
T$NRNI    000042         #6-72
T$NRPL    000005         #6-72
T$NRUL    000007         #6-72
T$NVR     000001         #6-72
T$RPRI    000040         #6-72
T$SVC     000034         #6-72
T$T5      000030         #6-72
T$T6      000032         #6-72
X$$MCB  = ******          6-77       6-77
ZF.COU  = 001000         #6-77
ZF.DDM  = 000001         #6-77
ZF.DIA  = 004000         #6-77
ZF.DLC  = 000002         #6-77
ZF.DVP  = 100000         #6-77
ZF.INI  = 040000         #6-77
ZF.KMX  = 000020         #6-77
ZF.LLC  = 000004         #6-77
ZF.LMC  = 000100         #6-77
ZF.MAN  = 020000         #6-77
ZF.MFL  = 000010         #6-77
```

```
                                                                   C  8
NTIMFE11S  CREATED BY  MACRO  ON 24-JUL-85 AT 15:52     PAGE 4
```

```
                                                .SBTTL  MAIN CONTROL

                                        ;+
                                        ; NTIGRQ - GET FUNCTION REQUEST AND DISPATCH
                                        ;
                                        ; THIS ROUTINE IS CURRENTLY CODED SUCH THAT FUNCTION REQUESTS TO
                                        ; NTINIT VIA "SEND DATA" MUST BE SYNCHRONOUS WITH NTINIT
                                        ; COMPLETIONS.  THAT IS, ANY REQUEST TO NTINIT IS ILLEGAL UNTIL
                                        ; THE PREVIOUS REQUEST HAS BEEN COMPLETED.
                                        ;
                                        ; INPUTS:
                                        ;       NONE
                                        ;
                                        ; OUTPUTS:
                                        ;       WAIT FOR A COMPLETED RCV DATA AND DISPATCH THE PROPER FUNCTION
                                        ;         ACCORDING TO "$NIFNC" (SET BY THE RCV DATA ROUTINE)
                                        ;       THE SEND DATA TO THE REQUESTING TASK IS ONLY DONE AFTER
                                        ;         THE FUNCTION IS COMPLETED.  THE RETURNED FUNCTION IS EITHER
                                        ;         POSITIVE (SUCCESS), ZERO (NO-OP), OR NEGATIVE (ERROR).
                                        ;-

 000006                                 NTIGRQ:
                                                .IF DF  R$$11S
 000006  032777  0000006 000426'                BIT     #FE.CEX,@FMASK          ;IS THE COMM EXFC LOADED ?
 000014  001410                                 BEQ     20$                     ; .. NO
                                                .ENDC

 000016  126727  000032' 000001                 CMPB    $NIFNC,#LS.CEX          ;IS IT A "SET SYS"?
 000024  001410                                 BEQ     30$                     ;BR IF YES

                                                .IF NDF R$$11S
                                                BIT     #FE.CEX,@FMASK          ;IS THE COMM EXEC LOADED ?
                                                BEQ     20$                     ; .. NO
                                                CMPB    $NIFNC,#LS.ON           ;IS IT "SET EXECUTOR ON"?
                                                BEQ     40$                     ;BR IF YES
                                                CMPB    $NIFNC,#LS.X25          ;IS IT SET MODULE X25-SERVER?
                                                BEQ     60$                     ;BR IF YES
                                                .ENDC

                                                .IF DF S$$BAS
                                                CMPB    $NIFNC,#LS.ON           ;IS IT "SET EXECUTOR STATE ON"?
                                                BEQ     40$                     ;IF EQ, YES
                                                .ENDC

 000026                                         EMSG$   03                      ;ILLEGAL FUNCTION
 000034  000452                                 BR      80$
 000036                                 20$:    EMSG$   04                      ;COMM EXEC NOT LOADED
 000044  000446                                 BR      80$
                                        ;
                                        ; CALLS TO FUNCTION ROUTINES.  THESE ROUTINES SHOULD RETURN WITH CARRY
                                        ; INDICATING SUCCESS/FAILURE, AND "$NIFNC" SET TO A PROPER RETURN
                                        ; COMPLETION CODE FOR THE FUNCTION REQUESTOR.
                                        ;
 000046                                 30$:    CALL    $NICEX                  ;FUNCTION - SET SYS
 000052  103443                                 BCS     80$                     ;BR IF FAILED
 000054  032767  010000  000032'                BIT     #LS.CXO,$NIFNC          ; IS IT EXECUTOR ON ALSO ?
 000062  001001                                 BNE     40$                     ; BR IF YES
```

```
                                          .SBTTL  PUTAST - PUT COMPLETION AST IN CALLER TASK'S AST QUEUE

                                  ;+
                                  ; PUTAST - PUT A COMPLETION AST IN THE CALLER TASK'S AST QUEUE
                                  ;
                                  ; INPUTS:
                                  ;         $NITRN=CALLING TASK'S NAME
                                  ;         $NIFNC=SUCCESS CODE
                                  ;         $NIAST=AST ADDRESS IN TASK
                                  ;         $NIPRM=PARAMETER TO RETURN TO TASK
                                  ;
                                  ; OUTPUTS:
                                  ;         AN AST IS QUEUED FOR THE CALLER.
                                  ;-

           000000                 $$$$ = 0                                ; ASSUME WE WILL ASSEMBLE THIS ROUTINE
           000001                         .IIF DF R$$11S  $$$$ = 1        ; WE DON'T NEED IT FOR 11S
                                          .IIF DF S$$BAS  $$$$ = 0        ; BUT WE DO NEED IT FOR SERVER BASE
                                                                          ; (EVEN THOUGH IT IS 11S BASED)

                                          .IF EQ  $$$$
                                  PUTAST:
                                          SWSTK$  30$                     ;; ENTER SYSTEM STATE
                                          MOV     #NTISEN+S.DATN,R3       ;; POINT TO TASK NAME
                                          CALL    @SRSTD                  ;; FIND CALLER'S TCB ADDRESS
                                          BCS     10$                     ;; IF CS, NOT THERE - TOO BAD
                                          MOV     R0,R5                   ;; ELSE, COPY TCB ADDRESS
                                          MOV     #7*2,R1                 ;; ELSE, SET AST BLOCK LENGTH
                                          CALL    @ALOCB                  ;; ALLOCATE SOME POOL
                                          BCC     20$                     ;; IF CC, GOT IT - PROCEED

                                  10$:    RETC    R0                      ;; RETURN CARRY SET
                                          BR      30$                     ;; ... AND RETURN

                                  20$:    MOV     R0,R2                   ;; COPY ADDRESS OF BLOCK ALLOCATED
                                          TST     (R0)+                   ;; BUMP OVER LINK WORD
                                          MOV     R1,(R0)+                ;; SET LENGTH OF BLOCK (FOR DEALLOCATION)
                                          MOV     #12*2,(R0)+             ;; SET STACK SPACE TO RESERVE FOR TASK
                                          MOV     $NIAST,(R0)+            ;; SET AST VIRTUAL ADDRESS FOR TASK
                                          MOV     #2,(R0)+                ;; SET PARAMETER COUNT
                                          MOV     $NIPRM,(R0)+            ;; SET PARAMETER #1 (USER SUPPLIED)
                                          MOV     $NIFNC,(R0)+            ;; SET PARAMETER #2 (COMPLETION STATUS)
                                          MOV     R5,R0                   ;; POINT TO TASK TCB
                                          MOV     R2,R1                   ;; POINT TO AST BLOCK
                                          CALL    @QASTT                  ;; QUEUE THE AST TO THE CALLING
                                  30$:    RETURN                          ;; BACK TO USER STATE AND RETURN

                                          .ENDC


           000000'                        .END    NTINIT
```

```
                                        .TITLE  NTITIM
                                        .IDENT  /V05.00/

                                ;
                                ; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ;       NTINIT - START COMM EXEC TIMERS
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  27-FEB-78
                                ;       VERSION 2.0 RELEASE
                                ;
                                ; 2.00  14-DEC-79
                                ;       DECNET-11M/S V3.0
                                ;       DECNET-11M-PLUS V1.0
                                ;
                                ; 3.00  16-APR-82
                                ;       DECNET-11M V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;

```

```
                                                .SBTTL  LOAD UMR'S

                                        ;+
                                        ; $NIUMR - INITIALIZE UMR'S ON AN 11/70 SYSTEM
                                        ;
                                        ; PLEASE NOTE THAT THIS CODE IS ALWAYS ASSEMBLED FOR AN 11/70 SYSTEM.
                                        ; IT SHOULD NEVER BE CALLED IF THERE ARE NO UMR'S TO LOAD.
                                        ;
                                        ; INPUTS:
                                        ;       R2 = UMR BLOCK ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       R2 = DESTROYED
                                        ;-

 000000  010346                         $NIUMR::MOV     R3,-(SP)                ;SAVE R3
 000002  010446                                 MOV     R4,-(SP)                ;SAVE R4
 000004  016203  000012                         MOV     M.BFVL(R2),R3           ;GET PARTITION ADDRESS
 000010  116204  000011                         MOVB    M.BFVH(R2),R4           ;...
 000014  016246  000002                         MOV     M.UMRA(R2),-(SP)        ;GET FIRST UMR ADDRESS
 000020  066216  000004                         ADD     M.UMRN(R2),(SP)         ;SET UPPER UMR LIMIT
 000024  016202  000002                         MOV     M.UMRA(R2),R2           ;POINT AT FIRST UMR
 000030  010322                         10$:    MOV     R3,(R2)+                ;LOAD THE ADDRESS TO MAP
 000032  010422                                 MOV     R4,(R2)+                ;  INTO A UMR
 000034  062703  020000                         ADD     #20000,R3               ;SET ADDRESS TO NEXT 4K WORDS
 000040  005504                                 ADC     R4                      ;...
 000042  020216                                 CMP     R2,(SP)                 ;MORE UMR'S TO LOAD?
 000044  103771                                 BLO     10$                     ;BR IF NO, STILL BELOW UPPER LIMIT
 000046  005726                                 TST     (SP)+                   ;DUMP UPPER LIMIT
 000050  012604                                 MOV     (SP)+,R4                ;RECOVER R4
 000052  012603                                 MOV     (SP)+,R3                ;RECOVER R3
 000054                                         RETURN
```

```
Symbol table

ARROWS  000503R         ERRILC= ****** GX       PRSKTB= ****** GX       $CLIOS= ****** GX       $NODNA= ****** GX
CR      000501R         ERRIPV  000433RG        PRSSTB= ****** GX       $CMAND= ****** GX       $OPRND= ****** GX
CRLF    000473R         ERRLID  000000RG        $BUFF = ****** GX       $COMP = ****** GX       $OPRN2= ****** GX
ERRCAL  000270RG        ERRLTY  000220RG        $CFCOM  000510RG        $ERROR  000506RG        $SWICH= ****** GX
ERRCHN  000162RG        ERRPRO  000113RG        $CFERR= ****** GX       $FLAGS= ****** GX       $TNUM = ****** GX
ERRCID  000044RG        ERRWLD  000341RG        $CFER1= ****** GX       $GTPLN= ****** GX       $VLVL = ****** GX
ERRFSP  000401RG        EX$ERR= 000002          $CIRCT= ****** GX       $INFO = ****** GX       $WILD = ****** GX
ERRGCL= ****** GX       IE.EOF= ****** GX       $CLERR  000760RG

. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        000766    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8484 Words  ( 34 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.04
SY:CFECOM.V2,[132,134]CFECOM/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]CFECOM
```

```
                                  ;
                                  ; ERRORS
                                  ;
 000772                           60$:    ERROR$  #ERRIWC         ; INVALID PARAMETER WITH KNOWN/WILDCARD
 001002  000411                           BR      90$

 001004                           70$:    ERROR$  #ERRIGR         ; ILLEGAL PARAMETER GROUPING
 001014  000404                           BR      90$

 001016                           80$:    ERROR$  #ERRPSI         ; NO PSI IN SYSTEM

 001026  000207                   90$:    RETURN

                                          .DSABL  LSB
```

```
                                              .SBTTL  $CFDSP - DISPATCH ON THE COMMAND CODE
                                      ;+
                                      ; *** - $CFDSP - DISPATCH ON THE COMMAND CODE
                                      ;
                                      ; INPUT:
                                      ;       $CMAND = COMMAND CODE TO DISPATCH ON
                                      ;
                                      ; OUTPUT:
                                      ;       CONTROL IS PASSED TO PROPER ROUTINE
                                      ;
                                      ; REGISTERS ALTERED:
                                      ;       R1
                                      ;
                                      ;-

000014                                $CFDSP::
000014  016701  000000G                       MOV     $CMAND,R1         ; GET FUNCTION CODE IN R1
000020  006301                                ASL     R1                ; CONVERT TO A WORD INDEX
000022  004771  177776'                       CALL    @DSPTBL-2(R1)     ; PROCESS THE COMMAND
000026  000207                                RETURN


        000001                                .END.
```

```
                .TITLE  CFEIMP - SYMBOL DEFINITIONS AND IMPURE STORAGE
                .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978, 1979, 1980, 1981, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; CFE - IMPURE AND PURE DATA
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  10-FEB-78
;       VERSION 3.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
NTIUMR  MACRO V05.03b Saturday 29-Jun-85 03:28
Table of contents

     5-   54     DATA
     6-   69     LOAD UMR'S
```

```
                                                .SBTTL  SET MODULE X25 SERVER

                                        ;+
                                        ; $NIX25 - SET STATE LOCAL ON
                                        ;
                                        ; REQUEST THE X25ACP TO START
                                        ;
                                        ;   *** NOTE: THIS MODULE SHOULD NOT BE ASSEMBLED FOR IAS SYSTEMS
                                        ;
                                        ; INPUTS:
                                        ;       X25ACP INSTALLED
                                        ;
                                        ; OUTPUTS:
                                        ;       THE DEVICE LIST IS SCANNED TO FIND THE X25ACP.  IF IT IS FOUND,
                                        ;       A VCB IS ALLOCATED FOR IT, AND IT IS REQUESTED TO RUN.
                                        ;       $NIRDB IS SET TO INDICATE SUCCESS (+) OR FAILURE (-) OR NO-OP (0).
                                        ;       CARRY IS SET ON FAILURE.
                                        ;-

000000                                  $NIX25::
000000                                          DIR$    #ASSIGN                 ;ASSIGN LUN 2 TO NW:
000006  103474                                  BCS     101$                    ;BR IF UNABLE TO ASSIGN LUN

                                                .IF     NDF,R$$MPL
000010  017700  000000G                         MOV     $HEADR,R0               ;GET TASK HEADER ADDRESS
000014  016005  000102                          MOV     H.LUN+4(R0),R5          ;GET NW: UCB ADDRESS
                                                .IFF
                                                SWSTK$  10$                     ;* ENTER KERNAL MODE
                                                SAVMAP                          ;* SAVE APR6 CONTENTS
                                                MOV     @SAHDB,@KSAR6           ;* MAP TO HEADER IN SECONDARY PO
                                                MOV     @SAHPT,R0               ;* GET THE HEADER POINTER
                                                MOV     H.LUN+4(R0),R5          ;* ADDRESS OF NS: UCB
                                                RESMAP                          ;* RESTORE APR6
                                                MOV     R5,14(SP)               ;* STORE VALUE IN USER R5
                                                RETURN                          ;* RETURN TO USER STATE
                                        10$:    .ENDC

000020  016504  000034                          MOV     U.VCB(R5),R4            ;IS THERE ALREADY A VCB ?
000024  001020                                  BNE     40$                     ; .. YES
000026  012701  000034                          MOV     #V$LEN,R1               ;VCB LENGTH
000032  005000                                  CLR     R0                      ;ASSUME FAILURE TO ALLOCATE VCB

000034                                          SWSTK$  30$                     ;* ENTER KERNAL MODE
000040                                          CALL    @ALOCB                  ;* TRY TO ALLOCATE FROM POOL
000044  103402                                  BCS     20$                     ;* BR IF FAILURE
000046  010066  000002                          MOV     R0,2(SP)                ;* SET USER MODE R0 VCB ADDRESS
000052                                  20$:    RETURN                          ;* BACK TO USER MODE

000054  012703  000206'                 30$:    MOV     #111$,R3                ; ASSUME ALLOCATION FAILURE
000060  010065  000034                          MOV     R0,U.VCB(R5)            ;STORE VCB ADDRESS
000064  001433                                  BEQ     70$                     ; .. ALLOCATION FAILURE
000066                                  40$:    DIR$    #LONACP                 ;REQUEST THE ACP
000074  103425                                  BCS     60$                     ; IF CS, SOMETHING'S WRONG

                                                .IF NDF R$$11S
000076  012702  062751                          MOV     #^RPLI,R2               ; GET RAD50 PROCESS NAME

```

```
                                        .SBTTL  LOCAL INITIALIZATION

                                ;+
                                ; NTILIN - LOCAL INITIALIZATION
                                ;
                                ; LOAD UMR'S AND VALIDATE CSR'S AS NEEDED
                                ;
                                ; THIS ROUTINE REQUIRES THE GLOBAL ADDRESSES ''$NIUMR'', AND ''$NICSR''
                                ; TO BE DEFINED.  ANY OF THESE ROUTINES MAY BE EXCLUDED FROM
                                ; THE TASK BUILD IF IT IS KNOWN THAT THEY WILL NOT BE CALLED (IF
                                ; EXCLUDED, THE SYMBOLS SHOULD BE GLOBALLY DEFINED AS FOLLOWS:
                                ; $NIUMR=177777, $NICSR=0).  ALSO, ON AN RSX11-S SYSTEM,
                                ; THE GLOBAL ''$NIDLI'' SHOULD BE DEFINED (ZERO IF MODULE IS EXCLUDED).
                                ;
                                ; INPUTS:
                                ;       $NTLPT          = ADDRESS (BIAS) OF NTL DATA CONTROL BLOCK
                                ;       BLOCK+2         = ADDRESS OF UMR CONTROL BLOCK (0 IF NONE)
                                ;       BLOCK+.CXCSR    = ADDRESS (BIAS) OF CSR CHECK LIST (0 IF NONE)
                                ;
                                ; OUTPUTS:
                                ;       ''$NIDLI'' IS CALLED TO INIT DOWN-LINE LOAD PARAMETERS (11S ONLY)
                                ;       IF BLOCK+2      IS NON-ZERO, ''$NIUMR'' IS CALLED
                                ;       IF BLOCK+.CXCSR IS NON-ZERO, ''$NICSR'' IS CALLED
                                ;       IF AN ERROR OCCURS IN ANY OF THESE, THE INITIALIZATION FAILED
                                ;       IF THE INITIALIZATION SUCCEEDED, THE TIMERS ARE STARTED
                                ;       CARRY IS SET IF INITIALIZATION FAILED
                                ;-

 000000                         NTILIN::CALL    FNDPDV                  ;FIND NSP PROCESSES (ECL & XPT)

 000004 017700  000000G                 MOV     @PDVTA,R0               ; GET PDV TABLE ADDRESS
 000010 017701  000000G                 MOV     @PDVNM,R1               ; GET NUMBER OF PDV'S
 000014 012002                  2$:     MOV     (R0)+,R2                ; GET NEXT PDV
 000016 032762  040000  000010          BIT     #ZF.INI,Z.FLG(R2)       ; PDV NEED INITIALIZATION ?
 000024 001426                          BEQ     4$                      ; IF NO - BRANCH
 000026 005762  000012                  TST     Z.PCB(R2)               ; IF PROCESS LOADED ?
 000032 001423                          BEQ     4$                      ; IF NO - BRANCH
 000034 016205  000016                  MOV     Z.DAT(R2),R5            ; GET DATA BASE ADDRESS
 000040 011503                          MOV     (R5),R3                 ; GET FIRST WORD
 000042 006003                          ROR     R3                      ; HAS PROCESS BEEN INITIALIZED ?
 000044 103415                          BCS     3$                      ; IF YES - BRANCH

 000046                                 SWSTK$  3$                      ; ENTER SYSTEM STATE
 000052 016202  000004                  MOV     Z.NAM(R2),R2            ;* GET PDV NAME
 000056                                 CALL    @PDVID                  ;* GET PDV INDEX
 000062 016746  000000'                 MOV     INIFUN,-(SP)            ;* GET INIT FUNCTION CODE
 000066 010603                          MOV     SP,R3                   ;* GET ADDRESS OF FUNCTION CODE
 000070                                 CALL    @LLCAL                  ;* DISPATCH TO PROCESS
 000074 005726                          TST     (SP)+                   ;* CLEAN UP STACK

 000076                                 RETURN

 000100 000241                  3$:     CLC
 000102 005301                  4$:     DEC     R1                      ; COUNT ONE LESS PDV
 000104 001343                          BNE     2$

                                        .IF     DF,R$$11S               ;*-CODE FOR AN RSX11S SYSTEM-*
```

```
     54                                           .SBTTL  DATA
     55
     56                                   ;
     57                                   ; LIBRARY MACROS
     58                                   ;
     59                                           .MCALL  DIR$,SVTK$,EMSG$,NTLER$,NHWDF$,SLTDF$,PDVDF$,SAVRG,RESRG
     60
     61 000000                                    NHWDF$                          ;HARDWARE REGISTER DEFINITIONS
     62 000000                                    SLTDF$                          ;SYSTEM LINE TABLE OFFSETS
     63 000000                                    PDVDF$                          ;PDV OFFSETS
     64
     65
     66                                   ;
     67                                   ; LOCAL (IMPURE) DATA
     68                                   ;
     69 000000                                    .PSECT  DATA,D
     70
     71                                   ;
     72                                   ; DIRECTIVES TO ALLOW PROGRAM TO TRAP ILLEGAL ADDRESS USAGE (FOR CSR CHECK)
     73                                   ;
     74 000000                            CSRENA: SVTK$   CSRTAB,1                ;TRAP USING "CSRTAB" TABLE
     75 000006                            CSRDIS: SVTK$                           ;DISABLE INTERNAL PROGRAM TRAPS
     76
     77 000014  000250'                   CSRTAB: .WORD   CSRTRP                  ;TRAP TO "CSRTRP" ON ILLEGAL ADDRESS
     78
     79                                   ;
     80                                   ; WORD TO HOLD SYSTEM LINE TABLE ADDRRESS
     81                                   ;
     82 000016                            CSRSLT: .BLKW   1                       ;SYSTEM LINE TABLE ADDRESS
     83
     84                                   ;
     85                                   ; ERROR MESSAGES
     86                                   ;
     87                                           .ENABL  LC
     88 000020                            NTLER$  <:>,20,NONE,$NIERR,$NERDN,,<Device * Offline>
     89                                           .DSABL  LC
     90
     91                                   ;
     92                                   ; PURE CODE SECTION
     93                                   ;
     94 000000                                    .PSECT
```

```
    134 000114  005721                  10$:    TST     (R1)+                           ;;POINT AT NODE NAME FIELD
    135 000116  005721                          TST     (R1)+                           ;;  ...
    136 000120  012100                          MOV     (R1)+,R0                        ;;IS THERE A NODE NAME PRESENT?
    137 000122  001411                          BEQ     15$                             ;;BR IF NO
    138
    139 000124  010063  000006                  MOV     R0,D$LNAM(R3)                   ;;SET IT IN CEXCM
    140 000130  012163  000010                  MOV     (R1)+,D$LNAM+2(R3)              ;;ALSO SECOND WORD
    141 000134  012163  000012                  MOV     (R1)+,D$LNAM+4(R3)              ;;ALSO THIRD WORD
    142 000140  062701  000012                  ADD     #10.,R1                         ;;SKIP PAST THE REST OF LOCAL NODE NAME
    143 000144  000402                          BR      20$                             ;;AND CHECK NEXT FIELD
    144 000146  062701  000016          15$:    ADD     #14.,R1                         ;;POINT AT NEXT FIELD
    145
    146 000152  062701  000020          20$:    ADD     #16.,R1                         ;;SKIP HOST NODE NAME FIELD
    147 000156  012100                          MOV     (R1)+,R0                        ;;GET HOST NODE ADDRESS
    148 000160  001414                          BEQ     21$                             ;;IF EQ, NONE
    149 000162  010046                          MOV     R0,-(SP)                        ;;SAVE HOST ADDRESS
    150 000164  042716  001777                  BIC     #^CAREA,(SP)                    ;;ISOLATE AREA
    151 000170  005726                          TST     (SP)+                           ;;AREA 0 ?
    152 000172  001005                          BNE     12$                             ;;IF NO - BRANCH
    153 000174  016346  000022                  MOV     D$HOST(R3),-(SP)                ;;SAVE VNP'D HOST ADDRESS
    154 000200  042716  001777                  BIC     #^CAREA,(SP)                    ;;ISOLATE AREA
    155 000204  052600                          BIS     (SP)+,R0                        ;;SET AREA
    156
    157 000206  010063  000022          12$:    MOV     R0,D$HOST(R3)                   ;;STORE HOST NODE ADDRESS
    158 000212  062701  000004          21$:    ADD     #4,R1                           ;;POINT AT DATE/TIME FIELD
    159 000216  105721                          TSTB    (R1)+                           ;;CHECK CENTURY IN DATE/TIME FIELD
    160 000220  001410                          BEQ     30$                             ;;IF EQ, NO TIME PARAMETERS TO HANDLE
    161 000222  012703  177764G                 MOV     #$TTNS-14,R3                    ;;POINT TO SYSCM DATE/TIME VECTOR
    162 000226  012702  000006                  MOV     #6,R2                           ;;NUMBER OF FIELDS TO COPY FROM PARAMETER
    163                                                                                 ;; MESSAGE TO CURRENT DATE/TIME VECTOR
    164 000232  112100                  22$:    MOVB    (R1)+,R0                        ;;GET NEXT DATE/TIME PARAMETER
    165 000234  010023                          MOV     R0,(R3)+                        ;;STORE IN DATE/TIME VECTOR IN SYSCM
    166 000236                                  SOB     R2,22$                          ;;LOOP THROUGH ALL THE PARAMETERS
    167
    168 000242                          30$:    RETURN                                  ;;FINISHED - RETURN TO USER STATE
    169
    170         000001                          .END
    295 000362                          350$:   EMSG$R  17                      ;RUN$ FAILURE
    296 000370                          360$:   EMSG$R  18                      ;PLB ALLOCATION FAILURE
    297 000376                          370$:   EMSG$R  11                      ;ROUTER PRIORITIES DATA BASE ALLOC. FAILURE
    298
```

```
                                        .SBTTL  DATA

                                ;
                                ; IF MEMORY MANAGEMENT IS IN EFFECT, ALWAYS ASSEMBLE CODE AS IF
                                ; IT WERE FOR THE 11/70
                                ;
              000000            .IIF    DF,M$$MGE,      M$$EXT=0        ;ALWAYS ASSEMBLE FOR THE 11/70

                                ;
                                ; LIBRARY MACROS
                                ;
                                        .MCALL  SLTDF$,PDVDF$,RETC,WSIG$S,CCBDF$,SAVRG,RESRG,XPDDB$,ADJDF$
                                        .MCALL  BIAS,MAP,SAVMAP,RESMAP,CTRDF$,SVCDF$,PHBDF$,PLBDF$
                                        .MCALL  MRKT$S,WTSE$S,EMSG$,DHBDF$,ECDDB$,CALLR

      000000                            DHBDF$                          ;DEFINE DECNET HOME BLOCK OFFSETS
      000000                            ECDDB$                          ;DEFINE ECL DATA BASE OFFSETS
      000000                            CCBDF$                          ;DEFINE CCB OFFSETS AND CONSTANTS
      000000                            CTRDF$                          ;DEFINE COUNTER BLOCK OFFSETS
      000000                            XPDDB$                          ;DEFINE TRANSPORTS DDB OFFSETS
      000000                            SLTDF$                          ;DEFINE SLT OFFSETS AND CONSTANTS
      000000                            SVCDF$                          ;DEFINE SVC OFFSETS
      000000                            PHBDF$                          ;DEFINE PSI HOME BLOCK OFFSETS
      000000                            PDVDF$                          ;DEFINE PDV OFFSETS AND CONSTANTS
      000000                            PLBDF$                          ;DEFINE PLB OFFSETS
      000000                            ADJDF$                          ;DEFINE ADJACENCY BLOCK OFFSETS

                                ;
                                ; CONSTANT DEFINITIONS
                                ;
              000004                    STRT = 4                        ;START STATUS FLAG FOR CCB
              000001                    TMEFN = 1                       ;TEMPORARY EVENT FLAG

                                ;
                                ; STATION ADDRESS
                                ;
      000000                    STAADD: .BLKW   1                       ;ADDRESS OF STATION TABLE FROM SLT
      000002                    STANUM: .BLKB   1                       ;STATION NUMBER
      000003                    CTLNUM: .BLKB   1                       ;CONTROLLER NUMBER
      000004  000000            ACTCIR: .WORD   0                       ;XPT CIRCUIT ACTIVE, ENDNODE ONLY


      000006                            .PSECT
```

```
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE              REFERENCES

ZF.MTM  = 000400            #6-77
ZF.MUX  = 000040            #6-77
ZF.PSE  = 002000            #6-77
ZF.SLI  = 010000            #6-77
ZF.TIM  = 000200            #6-77
ZF.X3P  = 000000            #6-77
ZS.ASN  = 100000            #6-77
ZS.BSY  = 140000            #6-77
Z.AVL     000014            #6-77
Z.DAT     000016            #6-77       8-285
Z.DSP     000000            #6-77       6-77
Z.FLG     000010            #6-77
Z.LEN   = 000016            #6-77
Z.LLN     000006            #6-77
Z.MAP     000020            #6-77
Z.NAM     000004            #6-77       7-179      7-223
Z.PCB     000012            #6-77
Z.SCH     000007            #6-77
$DBECL  = ******    GX       7-182      7-226
$ERR12  = ******    GX       7-193      7-237
$HEADR  = ******    GX       8-299
$NIMFE    000006 RG         #7-111
$SAVAL  = ******    GX       7-117
$$$ARG  = 000004            #7-113      7-113      7-113      7-113      7-113      7-113      7-113      7-113      7-113
                             7-113      7-113
```

```

                                        .IF NDF R$$11S
                                        BIT     #LS.CX5,$NIFNC          ; IS IT SET X25 ON ALSO?
                                        BNE     60$                     ; BR IF YES
                                        .ENDC

000064  000442                          BR      90$                     ; ELSE FINISH UP

000066  105777  000514'         40$:    TSTB    @LGSTT                  ;ARE ANY LOGGING SINKS ENABLED?
000072  001413                          BEQ     50$                     ;IF EQ, NO
000074                                  DIR$    #$NIMRK                 ;WAIT TWO SECONDS FOR EVL
000102                                  WTSE$S  #MRKTEF                 ;...
000114                                  DIR$    #$NIEVC                 ;TRY TO START EVENT COLLECTOR
000122                          50$:    CALL    $NILON                  ;FUNCTION - SET EXECUTOR ON
000126  103415                          BCS     80$                     ;BR IF ERROR

                                        .IF NDF S$$BAS                  ; DON'T PERMIT UNFIX ON SERVER BASE
                                          .IF DF R$$11S
000130  032767  020000  000032'         BIT     #LS.UNF,$NIFNC          ; IS THE UNFIX SUB-FUNCTION SET ?
000136  001415                          BEQ     90$                     ; .. NO
000140  017700  000446'                 MOV     @TKTCB,R0               ; GET OUR TCB ADDRESS
000144  042760  002000  000034          BIC     #T2.FXD,T.ST2(R0)       ; CLEAR THE FIXED BIT
000152  012760  131574  000010          MOV     #^R...,T.NAM+2(R0)      ; CHANGE INSTALLED NAME TO 'NTI...'
                                          .IFF  ;DF R$$11S
                                        BIT     #LS.CX5,$NIFNC          ; IS IT X25 STATE ON ALSO?
                                        BNE     60$                     ; BR IF YES
                                          .ENDC ;DF R$$11S
                                        .ENDC   ;NDF S$$BAS

000160  000404                          BR      90$                     ;ELSE EXIT WITH SUCCESS

                                        .IF NDF R$$11S
                                60$:    TSTB    @LGSTT                  ;ARE ANY LOGGING SINKS ENABLED?
                                        BEQ     70$                     ;IF EQ, NO
                                        DIR$    #$NIMRK                 ;WAIT TWO SECONDS FOR EVL
                                        WTSE$S  #MRKTEF                 ;...
                                        DIR$    #$NIEVC                 ;TRY TO START EVENT COLLECTOR
                                70$:    CALL    $NIX25                  ;FUNCTION - SET MODULE X25-SERVER
                                        BCS     80$                     ;BR IF ERROR
                                        BR      90$                     ;FINISH UP

                                        .ENDC

000162  012767  177777  000032' 80$:    MOV     #-1,$NIFNC              ;INDICATE NTINIT FUNCTION FAILED
000170  000402                          BR      100$                    ;AND FINISH UP
000172  005067  000032'         90$:    CLR     $NIFNC                  ;INDICATE SUCCESS

000176                          100$:   .IF NDF R$$11S
                                        DSAR$S                          ;DISABLE AST RECOGNITION
                                        CLR     $NIFLG                  ;CLR FLAGS AND BUSY INDICATION
                                        TST     $NIAST                  ;NOTIFY USERS VIA AST?
                                        BEQ     110$                    ;BR IF NO
                                        CALL    PUTAST                  ;ELSE NOTIFY USER VIA AST
                                        BR      120$                    ;AND CONTINUE

                                110$:   DIR$    #NTISEN                 ;SEND COMPLETION TO THE REQUESTOR
                                120$:   CALL    NTIRDX                  ;GET NEXT FUNCTION OR EXIT
```

```
                                                  C 10
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:09  Page 18-1
Symbol table

ABTIM   000456RG  002 CF.SYN= 000040        C.FLG2  000032        FS.LTM= 001000        LGSTT   000514RG  002
ALOCB   000436RG  002 CF.TIM= 000400        C.FNC   000010        FS.MNT= 004000        LLCAL   000524RG  002
AS.DEL= 000010        CF.TRN= 000100        C.LIN   000006        FS.MSN= 014000        LLCRS   000504RG  002
AS.EXT= 000004        CLKHD   000452RG  002 C.LNK   000000        FS.REA= 001000        LLCTA   000526RG  002
AS.RED= 000001        CMODE = 140000        C.MOD   000011        FS.RET= 000000        LN.CLO= 000000
AS.WRT= 000002        CM.CIR= 000002        C.NSP   000004        FS.REZ= 003000        LN.DUM= 000005
A$$CHK= 000000        CM.FMT= 100000        C.PRO   000042        FS.RLB= 002000        LN.LOA= 000004
A$$CPS= 000000        CM.HRD= 000002        C.RSV   000002        FS.RNG= 011000        LN.LOO= 000003
A$$PRI= 000000        CM.LIN= 000000        C.STA   000007        FS.RST= 000000        LN.OAU= 000003
A$$TRP= 000000        CM.LOO= 000001        C.STS   000012        FS.RTN= 001000        LN.OFF= 000001
A.IOC   000003        CM.XLO= 000004        C.URM   177776        FS.SET= 005000        LN.ON = 000000
A.LGTH= 000014        CP.DCF= 000040        C.XACP  000004        FS.SFC= 005000        LN.OOP= 000004
A.MPCT  000011        CP.HDL= 000007        C.XID   000035        FS.SFR= 006000        LN.OPE= 000001
A.PCB   000012        CP.PS = 177400        C.XLEN  000044        FS.SFS= 004000        LN.REF= 000002
A.PCBL  000000        CP.PSI= 000200        C.XPLI  000040        FS.SPW= 040000        LN.SER= 000002
A.PRI   000002        CP.XCF= 000100        C.XPT   000034        FS.STM= 000000        LN.STA= 000017
A.STAT  000010        CP.2FR= 000030        C.XSVC  000042        FS.STP= 002000        LN.SUB= 000360
A.TCB   000004        CS.ABO= 000100        C.XTC   000037        FS.STR= 001000        LN.TRI= 000006
A.TCBL  000006        CS.BRO= 000002        C.X25   000036        FS.TRM= 003000        LO.CON  000006 G
BUFUMP= 172354        CS.BUF= 000200        DEACB   000440RG  002 FS.WLB= 001000        LO.CSR  000006 G
CALLX   000472RG  002 CS.CES= 000002        DECPT   000516RG  002 FS.XKL= 002000        LO.CTL  000014 G
CB.CCB= 000002        CS.CHN= 000010        DEVHD   000460RG  002 FS.XOF= 010000        LO.LIN  000004 G
CB.DDM= 000040        CS.CMP= 000200        D$$BUG= 177514        FS.XON= 007000        LO.PAR  000010 G
CB.DLC= 000020        CS.DCR= 000400        D$$ISK= 000000        FS.ZER= 002000        LO.PRI  000010 G
CB.RDB= 000004        CS.DEF= 000004        D$$L11= 000001        F$$LVL= 000001        LO.SCR  000014 G
CB.SDB= 000010        CS.DEV= 000002        D$$YNC= 000000        G$$TPP= 000000        LO.STA  000015 G
CB.SLI= 000100        CS.DIS= 000040        D$$YNM= 000000        G$$TSS= 000000        LO.VCT  000012 G
CB.XLB= 000001        CS.ENA= 000001        EXEVEC  000424R   002 G$$TTK= 000000        LR.AST  000026 G
CCBGT   000502RG  002 CS.ENB= 000020        E$$XPR= 000000        G$$WRD= 000000        LR.CIR  000002 G
CC.LLC= 000200        CS.ERR= 100000        FC.CCP= 000020        HEADR   000444RG  002 LR.CTL  000004 G
CEACC   000510RG  002 CS.FTL= 001000        FC.CTL= 000006        IO.WVB= ****** GX     LR.EFN  000004 G
CE.ABO= 100362        CS.HCR= 000001        FC.KCP= 000016        I$$RAR= 000000        LR.LIN  000002 G
CE.DAO= 100346        CS.HFE= 002000        FC.KIL= 000004        I$$RDN= 000000        LR.PRM  000030 G
CE.DIS= 100366        CS.LST= 040000        FC.MAN= 000024        KISAR0= 172340        LR.PRO  000002 G
CE.ERR= 100370        CS.MTL= 004000        FC.MLD= 000026        KISAR6= 172354        LR.STS  000000 G
CE.ILN= 100350        CS.RNG= 000010        FC.PCT= 000030        KSAR6   000430RG  002 LR.TCB  000002 G
CE.LTO= 100356        CS.ROV= 000004        FC.PWR= 000022        K$$CNT= 177546        LR.UNT  000005 G
CE.MOP= 100372        CS.RSN= 010000        FC.RCE= 000002        K$$CSR= 177546        LS.AST= 100000 G
CE.NTE= 100361        CS.SHU= 000001        FC.RCP= 000014        K$$LDC= 000000        LS.CEX= 000001 G
CE.RTE= 100376        CS.SID= 000002        FC.TIM= 000010        K$$TPS= 000074        LS.CIR= 000010 G
CE.SRC= 100364        CS.STR= 000004        FC.XCP= 000012        LDPWF   000462RG  002 LS.CXO= 010000 G
CE.STP= 100352        CS.SUC= 000001        FC.XME= 000000        LD$LP = 000000        LS.CX5= 000400 G
CE.TME= 100354        CS.TMO= 020000        FE.CEX= ****** GX     LF.ACT= 100000        LS.DLM= 004000 G
CE.TMO= 100374        CS.XUR= 000004        FMASK   000426RG  002 LF.BRO= 000400        LS.ECH= 001000 G
CE.UNS= 100344        C$$CKP= 000000        FNDPDV  000204RG      LF.BWT= 000007        LS.FDX= 004000 G
CF.CHN= 000001        C$$ORE= 000400        FS.AST= 000000        LF.ENA= 002000        LS.HDX= 010000 G
CF.DDM= 000002        C$$RSH= 177564        FS.CIB= 002000        LF.LPB= 001000        LS.LIN= 000003 G
CF.DYN= 000004        C.ADD   000034        FS.CRA= 001000        LF.MDC= 000100        LS.LMC= 000007 G
CF.EIS= 000010        C.BID   000003        FS.DIS= 013000        LF.MFL= 004000        LS.NTI= 004000 G
CF.EOM= 000004        C.BUF   000014        FS.DVC= 001000        LF.MTP= 000020        LS.ON = 000001 G
CF.FRK= 100000        C.BUF1  000014        FS.FNB= 012000        LF.PAC= 000200        LS.OPT= 000400 G
CF.HDR= 000020        C.BUF2  000024        FS.EXI= 001000        LF.RDY= 040000        LS.PRO= 000002 G
CF.LB = 100000        C.CNT   000020        FS.GET= 006000        LF.REA= 010000        LS.PWF= 040000 G
CF.LIN= 000002        C.CNT1  000020        FS.HLT= 000000        LF.SER= 000040        LS.TOP= 002000 G
CF.LOG= 000020        C.CNT2  000030        FS.INI= 000000        LF.TIM= 000010        LS.UNF= 020000 G
CF.MDM= 000001        C.FLG   000022        FS.KIL= 000000        LF.UNL= 020000        LS.X25= 000012 G
CF.SOM= 000010        C.FLG1  000022        FS.LCL= 100000        LF.X2P= 000000        LS.11D= 004000 G
```

```
                                                  D 10
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:09  Page 18-2
```

```
     55                                         .SBTTL  DATA
     56
     57                                 ;
     58                                 ; LIBRARY MACROS
     59                                 ;
     60                                         .MCALL  NHWDF$,CLKDF$
     61
     62 000000                                  NHWDF$                          ;HARDWARE DEFINITIONS
     63 000000                                  CLKDF$                          ;DEFINE CLOCK QUEUE FNTRY OFFSETS
     64
     65                                 ;
     66                                 ; LOCAL DATA
     67                                 ;
     68
     69         000001                  T5      = 1                             ; 50 MSEC CLOCK BLOCK QUEUED
     70         000002                  T1      = 2                             ; 1 SEC CLOCK BLOCK QUEUED
     71
     72 000000  000000                  TIFLG:  .WORD   0                       ; TIMER BLOCK FLAG WORD
     73
     74
     75                                 ;
     76                                 ; START OF PURE CODE
     77                                 ;
     78 000002                                  .PSECT
```

```
    102         000001                          .END
```

```
CFECOM     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE            REFERENCES

ARROWS     000503 R      #4-104    5-152
CR         000501 R      #4-103    5-159
CRLF       000473 R      #4-102    5-147
ERRCAL     000270 RG     #4-98
ERRCHN     000162 RG     #4-96
ERRCID     000044 RG     #4-94
ERRFSP     000401 RG     #4-100
ERRGCL   = ******   GX    5-168
ERRILC   = ******   GX    5-140
ERRIPV     000433 RG     #4-101
ERRLID     000000 RG     #4-93
ERRLTY     000220 RG     #4-97
ERRPRO     000113 RG     #4-95
ERRWLD     000341 RG     #4-99
EX$ERR   = 000002        #4-92     4-93      4-94      4-95      4-96      4-97      4-98      4-99      4-100
                          4-101
IE.EOF   = ******   GX    5-165
PRSKTB   = ******   GX    5-138
PRSSTB   = ******   GX    5-139
$BUFF    = ******   GX    5-150
$CFCOM     000510 RG     #5-122
$CFERR   = ******   GX    5-160    5-168
$CFER1   = ******   GX    5-147    5-150     5-152     5-157     5-159
$CIRCT   = ******   GX   *5-130
$CLERR     000760 RG     #6-178
$CLIOS   = ******   GX    5-165
$CMAND   = ******   GX   *5-123
$COMP    = ******   GX   *5-126
$ERROR     000506 RG     #4-106   *5-140     5-160    *6-179
$FLAGS   = ******   GX   *5-128
$GTPLN   = ******   GX    5-141
$INFO    = ******   GX   *5-131
$NODNA   = ******   GX   *5-133   *5-134
$OPRND   = ******   GX   *5-124
$OPRN2   = ******   GX   *5-125
$SWICH   = ******   GX   *5-127
$TNUM    = ******   GX   *5-132
$VLVL    = ******   GX   *5-135
$WILD    = ******   GX   *5-129
```

```
                                                  .SBTTL  SETTRN - SET TRANSPORT CIRCUIT PARAMETERS

                                         ;+
                                         ; *** - SETTRN - SET TRANSPORT CIRCUIT PARAMETERS
                                         ;
                                         ; INPUT:
                                         ;        $OPRND = OPERANDS PRESENT
                                         ;
                                         ; OUTPUT:
                                         ;        CONTROL PASSED TO PROPER ROUTINE
                                         ;-

001030                                   SETTRN:
001030  032767  000000G 000000G                   BIT     #SW.SLT,$SWICH  ; CHANGE THE SLT/STATION DEFINITION ?
001036  001004                                    BNE     5$              ; IF NE, YES
001040  032767  000000G 000000G                   BIT     #OC.SLT,$OPRND  ; CHANGE THE SLT/STATION DEFINITION ?
001046  001402                                    BEQ     10$             ; IF EQ, NO
001050  004767  000000G                  5$:      CALL    $SESLT          ; CHANGE THE SLT DEFINITION
001054  032767  000000G 000000G          10$:     BIT     #OC.CNT,$OPRND  ; CHANGE THE CONTROLLER DEFINITION ?
001062  001403                                    BEQ     20$             ; IF EQ, NO
001064  004767  000000G                           CALL    $SECNT          ; ELSE, CHANGE THE CONTROLLER DEFINITION
001070  103405                                    BCS     40$             ; IF CS, ERROR
001072  032767  000000G 000000G          20$:     BIT     #OC.UNT,$OPRND ; CHANGE THE UNIT/STATION DEFINITION ?
001100  001402                                    BEQ     40$             ; IF EQ, NO
001102  004767  000000G                           CALL    $SEUNT          ; ELSE, CHANGE THE UNIT DEFINITION
001106  000207                           40$:     RETURN

                                                  .DSABL  LSB
```

```
                                                                                   C 15
CFEDSP - DIPATCH ROUTINE        MACRO V05.03b Saturday 29-Jun-85 04:55  Page 5-1
Symbol table

DSPTBL  000000R          $CFDSP  000014RG         $CFLIS= ****** GX        $CMAND= ****** GX        .EXIT = ****** GX
$CFDEF= ****** GX        $CFHLP= ****** GX        $CFPRG= ****** GX        $FIN   = ****** GX

. ABS.  000000     000   (RW,I,GBL,ABS,OVR)
        000030     001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 56 Words  ( 1 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:04.01
SY:CFEDSP.V2,[132,134]CFEDSP/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]CFEDSP
```

```
                                                                                   D 15
CFEDSP      CREATED BY  MACRO  ON 29-JUN-85 AT 04:55      PAGE 1
```

```
                                        ;****
                                        ; LIBRARY MACRO CALLS
                                        ;****

                                                .MCALL  QIOW$,FSRSZ$,FDBDF$,FDAT$A,FDRC$A,FDOP$A,FDOFF$
                                                .MCALL  GCLDF$,NSFDF$

```

```
                        .TITLE  NTIUMR
                        .IDENT  /V05.00/

                ;
                ;       COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
                ;       DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                ;
                ;
                ;       THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                ;       ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                ;       INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                ;       COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                ;       OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                ;       TRANSFERRED.
                ;
                ;       THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                ;       AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                ;       CORPORATION.
                ;
                ;       DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                ;       SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                ;
                ;       MODULE DESCRIPTION:
                ;
                ;               NTINIT - LOAD UNIBUS MAPPING REGISTERS FOR NETWORK BUFFER POOL
                ;
                ;       DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                ;
                ;       IDENT HISTORY:
                ;
                ;       1.00    27-FEB-78
                ;               VERSION 2.0 RELEASE
                ;
                ;       2.00    14-DEC-79
                ;               DECNET-11M/S V3.0
                ;               DECNET-11M-PLUS V1.0
                ;
                ;       3.00    16-APR-82
                ;               DECNET-11M V3.1
                ;               DECNET-11M-PLUS V1.1
                ;
                ;       4.00    7-NOV-83
                ;               DECNET-11M V4.0
                ;               DECNET-11M-PLUS V2.0
                ;
                ;       5.00    22-JUL-85
                ;               DECnet-11M/S V4.2
                ;               DECnet-11M-Plus V3.0
                ;               DECnet-Micro/RSX V1.0
                ;
                ;
```

```
                                                                   D 2
NTIX25 - NTINIT 'MODULE X25-SER MACRO V05.03b Saturday 29-Jun-85 03:29  Page 7-1
SET MODULE X25 SERVER

    162 000102                                 SWSTK$  45$                 ;; ENTER SYSTEM STATE
    163 000106                                 CALL    @PDVID              ;; GET PLI'S PDV INDEX
    164 000112  103006                         BCC     41$                 ;; BR IF PLI NOT IN SYSTEM
    165 000114                                 RETC    R0                  ;; RETURN CARRY SET
    166 000126  000402                         BR      42$                 ;; AND EXIT
    167 000130  010266  000004         41$:    MOV     R2,4(SP)            ;; STORE PDV INDEX IN USER R1
    168 000134                         42$:    RETURN                      ;; EXIT SYSTEM STATE
    169
    170 000136  103402                 45$:    BCS     50$                 ; BR IF PLI NOT IN SYSTEM
    171 000140                                 CALL    $NIMFE              ; DO THE MARK FOR ENABLE SCAN
    172                                        .ENDC   ; NDF R$$11S
    173
    174 000144  000241                 50$:    CLC                         ; ..SUCCESS - ALL DONE
    175 000146                                 RETURN
    176
    177 000150  012703  000156'        60$:    MOV     #90$,R3             ; DON'T FORGET TO DE-ALLOCATE THE VCB
    178 000154  000113                 70$:    JMP     (R3)                ; DE-ALLOCATE VCB OR PRINT ERROR MESSAGE
    179
    180 000156  010465  000034         90$:    MOV     R4,U.VCB(R5)        ;RESTORE ORIGINAL CONTENTS
    181 000162  001014                         BNE     121$                ; .. IT WAS NON-ZERO (?!?)
    182 000164  012701  000034                 MOV     #V$LEN,R1           ;VCB LENGTH
    183
    184 000170                                 SWSTK$  121$                ;* ENTER KERNEL MODE
    185 000174                                 CALLR   @DEACB              ;* DE-ALLOCATE THE VCB
    186
    187 000200                         101$:   EMSG$R  40                  ;ALUN$ FAILURE OR NW LOAD FAILURE
    188 000206                         111$:   EMSG$R  41                  ;VCB ALLOCATION FAILURE
    189 000214                         121$:   EMSG$R  42                  ;RUN$ FAILURE
    190                                        .ENDC
    191         000001                         .END
```

```
                                                                   E 2
NTIX25 - NTINIT 'MODULE X25-SER MACRO V05.03b Saturday 29-Jun-85 03:29  Page 7-2
Symbol table
```

```
000106  012700  000000G                   MOV     #$NIDLI,R0            ;GET DOWN-LINE LOAD INITIALIZER
000112  001401                            BEQ     5$                    ;BR IF NO SUCH MODULE
000114                                    CALL    (R0)                  ;PERFORM DOWN-LINE LOAD INIT
000116                            5$:     .ENDC                         ;*-END RSX11S CODE-*

                                          .IF DF  R$$MPL                                              ;[MP01]
                                          CALL    INILIN                ; REINIT HRC... IF RSX DATA STRUCTURES
                                                                        ; HAVE CHANGED                ;[MP01]
                                          .ENDC                                                       ;[MP01]

000116                                    SWSTK$  10$                   ;ENTER KERNEL MODE
000122  017777  000000G 000000G           MOV     @NTLPT,@KSAR6         ;;MAP TO THE NETLDR DATA BLOCK

                                          .IF      DF,M$$EXT
000130  016766  140002' 000006            MOV     .BASEB+2,6(SP)        ;;RETURN UMR BLOCK ADDRESS IN R2
                                          .ENDC

000136  016766  140000G 000010            MOV     .BASEB+.CXCSR,10(SP)  ;;RETURN CSR CHECK BLOCK ADDRESS IN R3
000144                                    RETURN                        ;;BACK TO USER MODE AT 10$
000146                            10$:
                                          .IF      DF,M$$EXT

000146  005702                            TST     R2                    ;IS THERE A UMR BLOCK ?
000150  001406                            BEQ     20$                   ;IF EQ, NO

                                            .IF DF R$$MPL
                                          BIT     #HF.UBM,@HFMSK        ;DOES THIS SYSTEM HAVE UMR'S?
                                          BEQ     20$                   ;BR IF NO
                                            .IFF            ; DF R$$MPL
000152  005777  000000G                   TST     @UMRPT                ;DOES THIS SYSTEM HAVE UMR'S?
000156  001403                            BEQ     20$                   ;BR IF NO
                                            .ENDC           ; DF R$$MPL

000160                                    CALL    $NIUMR                ;LOAD THE UMR'S
000164  000402                            BR      25$                   ; CONTINUE
                                          .ENDC

000166                            20$:    .IF DF  R$$MPL

                                          SWSTK$  25$                   ;;ENTER SYSTEM STATE IF M+
                                                                        ;;(READ/WRITE ACCESS TO LOW MEMORY)
                                          .IFTF   ;DF R$$MPL

000166  005077  000000G                   CLR     @PUMR                 ;INDICATE NO UMR'S

                                          .IFT    ;DF R$$MPL

                                          RETURN                        ;;RETURN TO USER STATE IF M+

                                          .ENDC

000172  012700  000000G           25$:    MOV     #$NICSR,R0            ;GET CSR CHECK ROUTINE ADDRESS
000176  001401                            BEQ     30$                   ;BR IF NO SUCH ROUTINE
000200                                    CALL    (R0)                  ;CHECK CSR VALIDITY

000202                            30$:
                                          .IF DF  R$$MPL                ; SPECIAL CASE HACK FOR NX     ;[MP01]
```

```
     96                                                  .SBTTL  CSR CHECK
     97
     98                                          ;+
     99                                          ; $NICSR - CHECK IF ALL CSR'S ARE ON-LINE
    100                                          ;
    101                                          ; CSR CHECK LIST FORMAT:
    102                                          ;
    103                                          ;       .WORD   LINK            ;LINK TO NEXT BLOCK IN LIST OR ZERO (EOL)
    104                                          ;       .REPT   15.             ;SERIES OF 2 WORD ENTRIES, 0 <= X <= 15.
    105                                          ;        .WORD   SLTA           ; SYSTEM LINE TABLE ADDRESS
    106                                          ;        .WORD   CSRA           ; CSR ADDRESS
    107                                          ;       .ENDR                   ;NUMBER OF ENTRIES MAY RANGE FROM 0. TO 15.
    108                                          ;
    109                                          ; INPUTS:
    110                                          ;       R3 = ADDRESS (BIAS) OF FIRST BLOCK IN CSR CHECK LIST (ZERO IF NONE)
    111                                          ;
    112                                          ; OUTPUTS:
    113                                          ;       ALL CSR'S IN LIST ARE CHECKED.
    114                                          ;       R3 IS DESTROYED.
    115                                          ;-
    116
    117 000000  005703                           $NICSR::TST    R3                      ;CLEAR CARRY, ANY CSR'S TO CHECK?
    118 000002  001521                                   BEQ    60$                     ;BR IF NO, FAST EXIT
    119 000004                                           SAVRG  <R0,R1,R2>              ;SAVE R0,R1,R2
    120 000012  012702  000017                           MOV    #17,R2                  ;SET INITIAL MAX COUNT
    121 000016  012701  140002                           MOV    #.BASEB+2,R1            ;R1 POINTS AT ENTRY TO CHECK
    122 000022  005000                           10$:    CLR    R0                      ;R0 WILL RETURN CSR ADDRESS OR ZERO
    123 000024                                           SWSTK$ 30$                     ;GO TO SYSTEM STATE
    124 000030  010377  000000G                          MOV    R3,@KSAR6               ;MAP TO CURRENT BLOCK IN LIST
    125 000034  005302                           15$:    DEC    R2                      ;COUNT DOWN NUMBER OF ENTRIES
    126 000036  002014                                   BGE    20$                     ;BR IF ANOTHER TO CHECK
    127 000040  016766  140000' 000010                   MOV    .BASEB,10(SP)           ;RESET USER R3 TO NEXT BLOCK
    128 000046  001423                                   BEQ    25$                     ;BR IF NO NEXT BLOCK, END OF LIST
    129 000050  016777  140000' 000000G                  MOV    .BASEB,@KSAR6           ;MAP TO NEXT BLOCK
    130 000056  012701  140002                           MOV    #.BASEB+2,R1            ;POINT AT FIRST ENTRY
    131 000062  012702  000017                           MOV    #17,R2                  ;RESET MAX COUNT
    132 000066  000762                                   BR     15$                     ;GO CHECK IT
    133
    134 000070  012167  000016'                  20$:    MOV    (R1)+,CSRSLT            ;PULL OUT THE SLT ADDRESS
    135 000074  001002                                   BNE    23$                     ;BR IF ENTRY IS NOT EMPTY
    136 000076  005721                                   TST    (R1)+                   ;SKIP SECOND WORD
    137 000100  000755                                   BR     15$                     ;GET NEXT ENTRY
    138
    139 000102  012166  000002                   23$:    MOV    (R1)+,2(SP)             ;SET USER R0 TO CSR ADDRESS
    140 000106  010166  000004                           MOV    R1,4(SP)                ;POINT USER R1 AT NEXT ENTRY
    141 000112  010266  000006                           MOV    R2,6(SP)                ;PUT CURRENT COUNT IN USER R2
    142
    143 000116                                   25$:    RETURN                         ;RETURN TO USER MODE AT "30$"
    144
    145 000120  005700                           30$:    TST    R0                      ;GOT CSR ADDRESS OR END OF LIST?
    146 000122  001445                                   BEQ    50$                     ;BR IF FOUND END OF LIST
    147
    148 000124  005046                                   CLR    -(SP)                   ;USE AS ARGUMENT FOR TRAP ROUTINE
    149 000126                                           DIR$   #CSRENA                 ;ENABLE ILLEGAL ADDRESS TRAPPING
    150 000134  005710                                   TST    (R0)                    ;CHECK THE CSR ADDRESS
    151 000136                                           DIR$   #CSRDIS                 ;DISABLE TRAPPING
    152 000144  006026                                   ROR    (SP)+                   ;WAS CSR ADDRESS ILLEGAL?
```

```
Symbol table

AREA  = 176000       D$LNAM  000006       D$$YNC= 000000       M$$CRX= 000000       P$$P45= 000000
A$$CHK= 000000       D$LNUM  000014       D$$YNM= 000000       M$$FCS= 000000       P$$WRD= 000000
A$$CPS= 000000       D$LST   000047       E$$XPR= 000000       M$$MGE= 000000       Q$$OPT= 000010
A$$PRI= 000000       D$MAXC  000064       F$$LVL= 000001       M$$NET= 000000       R$$DER= 000000
A$$TRP= 000000       D$MAXH  000066       G$$TPP= 000000       M$$OVR= 000000       R$$K11= 000001
BUFUMP= 172354       D$MAXV  000070       G$$TSS= 000000       N$$ACC= 000001       R$$SND= 000000
CMODE = 140000       D$MLL   000040       G$$TTK= 000000       N$$BUF= 000001       R$$11M= 000000
C$$CKP= 000000       D$MNOD  000041       G$$WRD= 000000       N$$LDV= 000001       R$$11S= 000001
C$$ORE= 000400       D$NA    000062       I$$RAR= 000000       N$$MCP= 000001       SWR   = 177570
C$$RSH= 177564       D$NBEA  000056       I$$RDN= 000000       N$$MLL= 000001       S$$WRG= 000000
D$AMXC  000072       D$NBRA  000054       KISAR0= 172340       N$$MOV= 000010       S$$YSZ= 007600
D$AMXH  000074       D$NEND= 000054       KISAR6= 172354       N$$NCT= 000001       TPS   = 177564
D$ANN   000000       D$NLN   000030       K$$CNT= 177546       N$$PEM= 000001       T$$KMG= 000000
D$BRPR  000102       D$NN    000060       K$$CSR= 177546       PIRQ  = 177772       T$$MIN= 000000
D$BRTM  000100       D$OUTT  000043       K$$LDC= 000000       PMODE = 030000       UBMPR = 170200
D$DELF  000045       D$RETF  000050       K$$TPS= 000074       PR0   = 000000       UISAR0= 177640
D$DELW  000046       D$RNN   000002       LD$LP = 000000       PR1   = 000040       UISAR1= 177642
D$END = 000104       D$RTMR  000076       L$$ASG= 000000       PR2   = 000100       V$$CTR= 001000
D$FNB   000034       D$SEG   000036       L$$DRV= 000000       PR3   = 000140       X$$DBT= 000000
D$HIOR  000024       D$SER   000032       L$$P11= 000001       PR4   = 000200       $DECPT= ****** GX
D$HOST  000022       D$SQRL  000052       L$$11R= 000000       PR5   = 000240       $NIDLI  000000RG
D$INAC  000044       D$$BUG= 177514       MPAR  = 172100       PR6   = 000300       $SYSIZ= ****** GX
D$INCT  000042       D$$ISK= 000000       MPCSR = 177746       PR7   = 000340       $TTNS = ****** GX
D$IPL   000051       D$$L11= 000001       M$$CRB= 000124       PS    = 177776       .BASEB= 140000
D$LID   000020

. ABS.  000104     000    (RW,I,GBL,ABS,OVR)
        000244     001    (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 11215 Words  ( 44 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:09.05
SY:NTIDLI11S.V2,[132,134]NTIDLI11S/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]V2,NTIDLI
```

```
                                .SBTTL  FNDPLI - GET PLI'S PDV INDEX
                        ;+
                        ;
                        ; FNDPLI - GET PDV INDEX FOR PLI PROCESS
                        ;
                        ; INPUTS:
                        ;       NONE
                        ;
                        ; OUTPUTS:
                        ;       R1 - PDV INDEX FOR PLI
                        ;
                        ;-

                                .IF NDF R$$11S

                        FNDPLI: SWSTK$  20$                     ;; ENTER SYSTEM STATE
                                MOV     #^RPLI,R2               ;; GET RAD50 PROCESS NAME
                                CALL    @PDVID                  ;; LOOK FOR IT
                                BCC     10$                     ;; BR IF FOUND IT

                                RETC    R0                      ;; RETURN CARRY SET
                                BR      20$                     ;; AND EXIT

                        10$:    MOV     R2,4(SP)                ;; STORE PDV INDEX IN USER R1

                        20$:    RETURN                          ;;

                                .ENDC

        000001                  .END
```

```
                                        .SBTTL  MARKED FOR ENABLE SCAN

                                ;+
                                ; $NIMFE - PERFORM MARKED-FOR-ENABLE SCAN
                                ;
                                ; INPUTS:
                                ;       ALL PROCESS/LINES MARKED FOR LOAD HAVE BEEN LOADED
                                ;
                                ; OUTPUTS:
                                ;       ALL LINES THAT ARE READY AND OWNED BY SPECIFIED LLC ARE ENABLED
                                ;       ALL REGISTERS ARE PRESERVED
                                ;-

000006                          $NIMFE::
                                        .IF NDF S$$BAS
000006                                  MRKT$S  #TMEFN,#1,#2            ; Wait 1 second in case system has
000032                                  WTSE$S  #TMEFN                  ;   been VNP'd and contains a DMV
                                        .ENDC

000044                                  CALL    $SAVAL                  ;SAVE ALL REGISTERS
000050                                  CALL    FNDPDV                  ;FIND ECL/XPT DATA STRUCTURES
000054  005067  177724                  CLR     ACTCIR                  ;INIT CIRCUIT ENABLED FLAG
000060  017746  000000G                 MOV     @SLTNM,-(SP)            ;SET NUMBER OF SYSTEM LINES
000064  017702  000000G                 MOV     @LLCTA,R2               ;POINT TO FIRST REVERSE MAPPING
000070  017704  000000G                 MOV     @SLTMA,R4               ;POINT TO SLT INDEX TABLE
000074  012405                  ENAC:   MOV     (R4)+,R5                ;GET POINTER TO SYSTEM LINE TABL
000076  005067  177676                  CLR     STAADD                  ;ASSUME NOT MULTIPOINT LINE
000102  032765  040000  000000          BIT     #LF.RDY,L.FLG(R5)       ;IS EVERYTHING READY (DDM,DLC,ETC.)?
000110  001002                          BNE     1$                      ;IF YES - ENABLE IT
000112                                  CALLR   60$                     ;CONTINUE

000116  116500  000003          1$:     MOVB    L.DLC(R5),R0            ;GET PDV INDEX OF DDM
000122  067700  000000G                 ADD     @PDVTA,R0               ;POINT TO CORRECT PDV
000126  011000                          MOV     (R0),R0                 ;GET PDV ADDRESS
000130  032765  100000  000000          BIT     #LF.ACT,L.FLG(R5)       ;IS LINE ALREADY ACTIVE?
000136  001412                          BEQ     5$                      ;IF NO - BRANCH

000140  032765  000400  000000          BIT     #LF.BRO,L.FLG(R5)       ;BROADCAST LINE ?
000146  001404                          BEQ     4$                      ;IF NO - BRANCH
000150  132765  000200  000022          BITB    #SF.ACT,L.MPF(R5)       ;XPT CHANNEL ACTIVE ?
000156  001402                          BEQ     5$                      ;IF NO - BRANCH

000160                          4$:     CALLR   60$                     ;CONTINUE

000164                          5$:
                                        .IF DF  R$$MPL

                                        BIT     #ZF.PSE,Z.FLG(R0)       ;DLM CIRCUIT ?
                                        BNE     20$                     ;IF YES - BRANCH

                                        MOV     L.KRBA(R5),R3           ;GET POINTER TO KRB
                                        BIT     #KS.OFL,K.STS(R3)       ;IS THE CONTROLLER OFF-LINE?
                                        BNE     4$                      ;YES ... CAN'T ENABLE IT
                                        TSTB    @NCPU                   ;ARE WE RUNNING ON A MULTI-PROCESSOR?
                                        BEQ     20$                     ;NO ... WE CAN ENABLE THE LINE
                                        BIT     K.URM(R3),@URMST        ;IS THE REQUIRED UNIBUS ACTIVE?
                                        BEQ     4$                      ;NO ... WE CAN'T ENABLE THE LINE
```

```
                                                                 D  8
NTIMFE11S  CREATED BY  MACRO  ON 24-JUL-85 AT 15:52      PAGE 5

MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME      REFERENCES

ADJDF$          #6-65       6-79
BIAS            #6-66
CALL             7-117      7-118      7-193      7-198      7-237      7-241      8-288      8-289
CALLR           #6-67       7-127      7-140      7-249      8-297
CCBDF$          #6-65       6-71
CTRDF$          #6-66       6-72
DHBDF$          #6-67       6-69
DIR$            #7-113      7-113     #7-114      7-114     #8-303      8-303
ECDDB$          #6-67       6-70
EMSG$           #6-67       7-193      7-237
MAP             #6-66
MOV$            #7-113      7-113      7-113      7-113      7-113     #7-114      7-114
MRKT$S          #6-67       7-113
PDVDF$          #6-65       6-77
PHBDF$          #6-66       6-76
PLBDF$          #6-66       6-78
RESMAP          #6-66
RESRG           #6-65       8-307
RETC            #6-65       8-299
RETURN           7-253      8-300      8-308
RVP$            #7-113      7-113
SAVMAP          #6-66
SAVRG           #6-65       8-269
SLTDF$          #6-65       6-74
SVCDF$          #6-66       6-75
SWSTK$           8-288
WSIG$S          #6-65       8-303
WTSE$S          #6-67       7-114
XPDDB$          #6-65       6-73
```

```
                                                                 E  8
**FILE**ID**NTINIT
```

```
    448                                          ENAR$S                          ;ENABLE AST'S AGAIN
    449                                          CALLR   NTIGRQ                  ;PROCESS NEXT REQUEST
    450                                          .IFF    ;NDF R$$11S
    451
    452                                            .IF DF S$$BAS
    453                                          DSAR$S                          ;DISABLE AST RECOGNITION
    454                                          TST     $NIAST                  ;NOTIFY USERS VIA AST?
    455                                          BEQ     110$                    ;BR IF NO
    456                                          CALL    PUTAST                  ;ELSE NOTIFY USER VIA AST
    457                                          BR      120$                    ;AND CONTINUE
    458
    459                                  110$:   DIR$    #NTISEN                 ;SEND COMPLETION TO THE REQUESTOR
    460                                  120$:   ENAR$S                          ;ENABLE AST'S AGAIN
    461                                            .ENDC
    462
    463 000176                                   EXIT$S
    464
    465                                          .ENDC
    466
```

```
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:09  Page 18-2
Symbol table

LX.CEX= 000004 G       N$$PEM= 000001             R.QSPN= 000006             TS.RUN= 004000             T3.RST= 000400
LX.LIN= 000006 G       PDVID   000522RG       002 R.QSPR= 000012             TS.STP= 001000             T3.SLV= 002000
LX.PRO= 000005 G       PDVNM   000520RG       002 R.QSTN= 000002             T$$KMG= 000000             T3.SWS= 000002
L$$ASG= 000000         PDVTA   000530RG       002 R.VXBA= 000006             T$$MIN= 000000             UBMPR = 170200
L$$DRV= 000000         PIRQ  = 177772             R.VXTN= 000002             T.ACTL  000052             UISAR0= 177640
L$$P11= 000001         PMODE = 030000             SF.ACT= 000200             T.ASTL  000016             UISAR1= 177642
L$$11R= 000000         PR0   = 000000             SF.ENA= 000100             T.ATT   000062             UMRPT   000432RG       002
L.COST  000015         PR1   = 000040             SF.LPB= 000004             T.CPCB  000004             VECLN = 000043 G
L.CTL   000012         PR2   = 000100             SF.MFL= 000040             T.DPRI  000040             VF.LIB= 000001
L.CVA   177776         PR3   = 000140             SF.PAC= 000020             T.EFLG  000022             V$$CTR= 001000
L.DDM   000002         PR4   = 000200             SF.REA= 000010             T.IOC   000003             V.ADD   000006
L.DDS   000004         PR5   = 000240             SF.SER= 000001             T.LBN   000041             V.BIAS  000004
L.DLC   000003         PR6   = 000300             SF.SVC= 000002             T.LDV   000044             V.FLG   000002
L.DLM   000006         PR7   = 000340             SF.UNL= 000040             T.LNK   000000             V.LEN   000012
L.DLS   000010         PS    = 177776             SLTMA   000464RG       002 T.MXSZ  000050             V.LNK   000000
L.FLG   000000         PSIPT   000506RG       002 SLTNM   000466RG       002 T.NAM   000006             V.PDV   000004
L.KRBA  000016         PS.APR= 000007             SPARE = 000010             T.OFF   000066             V.SIZ   000010
L.LEN = 000022         PS.CHK= 010000             SP.EIP= 000001             T.PCB   000046             X$$DBT= 000000
L.MPF   000022         PS.CKP= 040000             SP.ENB= 000002             T.PRI   000002             ZF.COU= 001000
L.NMST  000020         PS.CKR= 020000             SP.LOG= 000004             T.RCVL  000012             ZF.DDM= 000001
L.NSTA  000014         PS.COM= 000200             SRSTD   000434RG       002 T.RRFL  000072             ZF.DIA= 004000
L.OWNR  000021         PS.DEL= 000010             SWR   = 177570             T.SAST  000054             ZF.DLC= 000002
L.UNT   000013         PS.DRV= 000020             S$$WRG= 000000             T.SRCT  000071             ZF.DVP= 100000
MPAR  = 172100         PS.FXD= 004000             S$$YSZ= 007600             T.STAT  000032             ZF.INI= 040000
MPCSR = 177746         PS.LIO= 001000             S.BMSK  177776             T.ST2   000034             ZF.KMX= 000020
MRKTEF= 000001         PS.NSF= 000400             S.BMSV  177774             T.ST3   000036             ZF.LLC= 000004
M$$CRB= 000124         PS.OUT= 100000             S.CON   000010             T.TCBL  000030             ZF.LMC= 000100
M$$CRX= 000000         PS.PER= 002000             S.COST  000001             T.TIO   000057             ZF.MAN= 020000
M$$FCS= 000000         PS.PIC= 000100             S.CSR   000012             T.TKSZ  000060             ZF.MFL= 000010
M$$MGE= 000000         PS.SYS= 000040             S.CTM   000006             T.UCB   000026             ZF.MTM= 000400
M$$NET= 000000         PUMR    000442RG       002 S.DMCS  000020             T1SCL   000512RG       002 ZF.MUX= 000040
M$$OVR= 000000         P$$P45= 000000             S.FLG   000000             T100C   000474RG       002 ZF.PSE= 002000
M.BFVH  000011         P$$WRD= 000000             S.FRK   000016             T100Q   000476RG       002 ZF.SLI= 010000
M.BFVL  000012         P.BLKS  000016             S.ITM   000007             T2.ABO= 000100             ZF.TIM= 000200
M.KTAE= 000010         P.BUSY  000024             S.LEN   000004             T2.AST= 100000             ZF.X3P= 000000
M.KTEF= 000002         P.IOC   000003             S.LHD   000000             T2.CAF= 000400             ZS.ASN= 100000
M.KTMC= 000004         P.LNK   000000             S.NMST  000002             T2.CHK= 020000             ZS.BSY= 140000
M.KTUN= 000006         P.MAIN  000012             S.OWNR  000003             T2.CKD= 010000             Z.AVL   000014
M.LGTH= 000014         P.NAM   000004             S.PKT   000014             T2.DST= 040000             Z.DAT   000016
M.LNK   000000         P.OWN   000026             S.PRI   000004             T2.FXD= 002000             Z.DSP   000000
M.UMRA  000002         P.PRI   000002             S.RCNT  177772             T2.HLT= 000200             Z.FLG   000010
M.UMRN  000004         P.REL   000014             S.ROFF  177773             T2.REX= 001000             Z.LEN = 000016
M.UMVH  000010         P.SIZE  000016             S.STS   000011             T2.SEF= 004000             Z.LLN   000006
M.UMVL  000006         P.STAT  000030             S.VCT   000005             T2.SPN= 000004             Z.MAP   000020
NTIERB  000137R    002 P.SUB   000010             TKPS    000454RG       002 T2.STP= 000020             Z.NAM   000004
NTIERN  000124R    002 P.SWSZ  000022             TKTCB   000446RG       002 T2.WFR= 000001             Z.PCP   000012
NTIGRQ  000006R        P.TCB   000026             TK100   000500RG       002 T3.ACP= 100000             Z.SCH   000007
NTINIT  000000R        P.WAIT  000020             TPS   = 177564             T3.CAL= 000100             $ABTIM= ****** GX
NTIRC   000016R    002 QASTT   000450RG       002 TS.BLK= 171700             T3.CLI= 001000             $ALOCB= ****** GX
NTLPT   000470RG   002 Q$$OPT= 000010             TS.CKP= 000200             T3.GFL= 000010             $CALLX= ****** GX
N$$AC   000001         R$$DER= 000000             TS.CKR= 000100             T3.MCR= 004000             $CBDMG= ****** GX
N$$BU   000001         R$$K11= 000001             TS.EXE= 100000             T3.NET= 000020             $CCBGT= ****** GX
N$$LD   000001         R$$SND= 000000             TS.FLD= 002000             T3.NSD= 000200             $CEACC= ****** GX
N$$MC = 000001         R$$11M= 000000             TS.MSG= 020000             T3.PMD= 040000             $CLKHD= ****** GX
N$$ML   000001         R$$11S= 000001             TS.NRP= 010000             T3.PRV= 010000             $C5TA = ****** GX
N$$M    000010         R.QSGC= 000015             TS.OUT= 000400             T3.REM= 020000             $DBECL  000004RG       002
N$$N  = 000001         R.QSPC= 000014             TS.RDN= 040000             T3.ROV= 000040             $DBXPT  000006RG       002
```

```
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:09  Page 18-3
```

```
                                                        .SBTTL  START CEX TIMERS

                                                ;+
                                                ; $NITIM - START CEX TIMERS IF NOT ALREADY STARTED
                                                ;
                                                ; INPUTS:
                                                ;       NONE
                                                ;
                                                ; OUTPUTS:
                                                ;       IF THE TIMERS ARE ALREADY IN THE CLOCK QUEUE, THEN THIS IS A
                                                ;         NO-OP, ELSE THE TIMERS ARE STARTED.
                                                ;       R0 IS DESTROYED.
                                                ;-

 000002  016700  000000G                        $NITIM::MOV     CLKHD,R0                ;POINT AT THE CLOCK QUEUE

 000006                                                 SWSTK$  30$                     ;* ENTER SYSTEM STATE

 000012                                                 MTPS    #PR7                    ;;;* DISABLE INTERRUPTS
 000020  011000                                 10$:    MOV     (R0),R0                 ;;;* POINT AT NEXT BLOCK
 000022  001416                                         BEQ     20$                     ;;;* IF EQ, END
 000024  026700  000000G                                CMP     T100C,R0                ;;;* ENTRY FOR 50 MSEC TIMER?
 000030  001004                                         BNE     12$                     ;;;* IF NO - BRANCH
 000032  052767  000001  177740                         BIS     #T5,TIFLG               ;;;* IF YES - SET 50 MSEC FLAG
 000040  000767                                         BR      10$                     ;;;* CONTINUE

 000042  026700  000000G                        12$:    CMP     T1SCL,R0                ;;;* ENTRY FOR 1 SEC TIMER?
 000046  001364                                         BNE     10$                     ;;;* IF NE, NO - CHECK NEXT BLOC
 000050  052767  000002  177722                         BIS     #T1,TIFLG               ;;;* IF YES - SET 1 SEC FLAG
 000056  000760                                         BR      10$                     ;;;* CONTINUE

 000060                                         20$:    MTPS    #0                      ;;;* RE-ENABLE INTERRUPTS
 000064  032767  000001  177706                         BIT     #T5,TIFLG               ;* START 50 MSEC TIMER ?
 000072  001002                                         BNE     25$                     ;* IF NO - BRANCH
 000074                                                 CALL    T50IN                   ;* START 50 MSEC TIMER (LOCAL CODE)
 000100  032767  000002  177672                 25$:    BIT     #T1,TIFLG               ;* START 1 SEC TIMER ?
 000106  001002                                         BNE     30$                     ;* IF NO - BRANCH
 000110                                                 CALL    T1SIN                   ;* START 1 SEC TIMER (LOCAL CODE)

 000114                                         30$:    RETURN                          ;* RETURN TO USER AND THEN TO CALLER
```

```
NTIUMR  MACRO V05.03b Saturday 29-Jun-85 03:33  Page 8-1
Symbol table

A$$CHK= 000000        G$$WRD= 000000        M$$OVR= 000000        P$$P45= 000000        S.CSR   000012
A$$CPS= 000000        I$$RAR= 000000        M.BFVH  000011        P$$WRD= 000000        S.CTM   000006
A$$PRI= 000000        I$$RDN= 000000        M.BFVL  000012        Q$$OPT= 000010        S.DMCS  000020
A$$TRP= 000000        K$$CNT= 177546        M.LGTH= 000014        R$$DER= 000000        S.FRK   000016
C$$CKP= 000000        K$$CSR= 177546        M.LNK   000000        R$$K11= 000001        S.ITM   000007
C$$ORE= 000400        K$$LDC= 000000        M.UMRA  000002        R$$SND= 000000        S.LHD   000000
C$$RSH= 177564        K$$TPS= 000074        M.UMRN  000004        R$$11M= 000000        S.PKT   000014
D$$BUG= 177514        LD$LP = 000000        M.UMVH  000010        R$$11S= 000001        S.PRI   000004
D$$ISK= 000000        L$$ASG= 000000        M.UMVL  000006        SPARE = 000010        S.RCNT  177772
D$$L11= 000001        L$$DRV= 000000        N$$ACC= 000001        SP.EIP= 000001        S.ROFF  177773
D$$YNC= 000000        L$$P11= 000001        N$$BUF= 000001        SP.ENB= 000002        S.STS   000011
D$$YNM= 000000        L$$11R= 000000        N$$LDV= 000001        SP.LOG= 000004        S.VCT   000005
E$$XPR= 000000        M$$CRB= 000124        N$$MCP= 000001        S$$WRG= 000000        T$$KMG= 000000
F$$LVL= 000001        M$$CRX= 000000        N$$MLL= 000001        S$$YSZ= 007600        T$$MIN= 000000
G$$TPP= 000000        M$$FCS= 000000        N$$MOV= 000010        S.BMSK  177776        V$$CTR= 001000
G$$TSS= 000000        M$$MGE= 000000        N$$NCT= 000001        S.BMSV  177774        X$$DBT= 000000
G$$TTK= 000000        M$$NET= 000000        N$$PEM= 000001        S.CON   000010        $NIUMR  000000RG


. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000056    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 9373 Words  ( 37 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.45
SY:NTIUMR11S.V2,[132,134]NTIUMR11S/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]V2,NTIUMR
```

```
NTIUMR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:33      PAGE 1
```

```
CFECOM     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 2
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

EPRINT           #4-74       5-147       5-150       5-152       5-157       5-159
ERROR$           #4-63       5-160       5-168
PRINT            #4-58
```

```
**FILE**ID**CFEDEF
```

```
                                                .SBTTL  SETSVC - SET SVC PARAMETERS
                                                .SBTTL  SETPVC - SET PVC PARAMETERS
                                        ;+
                                        ; *** - SETSVC - SET SVC PARAMETERS
                                        ; *** - SETPVC - SET PVC PARAMETERS
                                        ;
                                        ; INPUT:
                                        ;       $OPRND = OPERANDS PRESENT
                                        ;
                                        ; OUTPUT:
                                        ;       CONTROL PASSED TO PROPER ROUTINE
                                        ;-
                                                .ENABL  LSB

001110  004767  000000G                 SETPVC: CALL    $SEPVC          ; SET PVC PARAMETERS
001114  000402                                  BR      5$              ; JOIN COMMON CODE
001116  004767  000000G                 SETSVC: CALL    $SESVC          ; SET SVC PARAMETERS
001122  103423                          5$:     BCS     10$             ; IF CS, FAILURE
                                        ;
                                        ; DLM CIRCUITS: SET UNIT AND SLT PARAMETERS. TEMPORARILY
                                        ; TURN OFF ANY LINE OPTIONS ON THE UNT$DF AND SLT$DF (SOME
                                        ; OF WHICH, UNFORTUNATELY, HAVE THE SAME VALUES OF CIRCUIT
                                        ; OPTIONS),
                                        ;
001124  016746  000000G                         MOV     $OPRND,-(SP)    ; SAVE OPERATIONS FLAGS
001130  042767  000000C 000000G                 BIC     #^C<OC.CST!OC.HTM!OC.STA>,$OPRND
001136  032767  000000G 000000G                 BIT     #OC.UNT,$OPRND  ; ANY UNIT PARAMETERS TO SET ?
001144  001402                                  BEQ     6$              ; IF EQ, NO
001146  004767  000000G                         CALL    $SEUNT          ; SET SVC COST
001152  032767  000000C 000000G 6$:             BIT     #OC.STA!OC.HTM,$OPRND   ; SET STATE OR HELLO TIMER?
001160  001402                                  BEQ     7$              ; IF EQ, NO
001162  004767  000000G                         CALL    $SESLT          ; SET SVC TRIB STATE AND HELLO TIMER
001166  012667  000000G                 7$:     MOV     (SP)+,$OPRND    ; RESTORE OPERATION FLAGS
001172  000207                          10$:    RETURN

                                                .DSABL  LSB

```

```
CFEDSP     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE             REFERENCES

DSPTBL    000000 R        #4-62         5-87
$CFDEF  = ******   GX      4-62
$CFDSP    000014 RG       #5-84
$CFHLP  = ******   GX      4-64
$CFLIS  = ******   GX      4-63
$CFPRG  = ******   GX      4-67
$CMAND  = ******   GX      5-85
$FIN    = ******   GX      4-65
.EXIT   = ******   GX      4-66
```

FILEID**CFEHLP

```
     64                                              .SBTTL  SYMBOL DEFINITIONS
     65                                      ;
     66                                      ; EQUATED SYMBOLS
     67                                      ;
     68          000001                              LUN1 = 1                  ; GCL LUN
     69          000002                              LUN2 = 2                  ; TERMINAL LUN
     70          000003                              LUN3 = 3                  ; ERROR LUN
     71          000004                              LUN4 = 4                  ; OUTPUT FILE LUN
     72          000005                              LUN5 = 5                  ; INPUT FILE LUN
     73          000006                              LUNX = 6                  ; SCRATCH LUN
     74          000764                              BUFSIZ =  500.            ; BUFFER SIZE IN BYTES
     75          000036                              MAXHOP == 30.             ; MAXIMUM NUMBER OF HOPS ALLOWED
     76          001776                              MAXCST == 1022.           ; MAXIMUM COST ALLOWED
     77          000077                              MAXARE == 63.             ; MAXIMUM AREA NUMBER
     78          001777                              MAXNN  == 1023.           ; MAXIMUM NUMBER OF NODES ALLOWED
     79          000031                              MXCSTL == 25.             ; MAXIMUM COST FOR A LINE
     80          000100                              $MXSTA == 64.             ; MAXIMUM NUMBER OF STATIONS ALLOWED
     81          000100                              $MXNLT == 64.             ; MAXIMUM # OF LINE TABLES AT END OF PROCESS
     82                                                                        ; LARGEST VALUE ALLOWED IS 255., FORMULA:
     83                                                                        ; (process size + (NLT * $$LGTH)) <= (4K WORDS)
     84          000100                              $MXNCT == 64.             ; MAXIMUM # OF CONTROLLER TABLES AT END OF DDM
     85                                                                        ; PROCESS. SIMILAR TO $MXNLT.
     86
     87 000000                                       NSFDF$  ,<=>,LIST         ; DEFINE CNFIG$ BITS

                 000001                      NF.ACC==1                         ; ACCESS CONTROL SUPPORT
                 000002                      NF.EVT==2                         ; EVENT LOGGING SUPPORT
                 000004                      NF.LLI==4                         ; LOGICAL LINK INTEGRITY SUPPORT
                 000010                      NF.SWP==10                        ; TASKS SWAPABLE WITH OUTSTANDING I/O
                 000020                      NF.SMC==20                        ; SPAWN MULTIPLE COPIES OF THE TASK
                 000040                      NF.BCP==40                        ; BACKWARD COMPATIBILITY (PREVIOUS VERSION) SUPPORT
                 000100                      NF.LV2==100                       ; LEVEL 2 ROUTING NODE
                 000200                      NF.END==200                       ; SYSTEM IS AN END NODE
                 000400                      NF.SLI==400                       ; SYSTEM LEVEL INTERFACE INCLUDED
                 040000                      NF$EVT==40000                     ; EVENT LOGGING IS ENABLED
                 100000                      NF$ACC==100000         ; ACCESS VERIFICATION IS ENABLED
                                                     .IF NB LIST
```

```
                                  .SBTTL  DATA

                           ; LIBRARY MACROS
                           ;
                                  .MCALL  SCBDF$

        000000                    SCBDF$                  ; UMR BLOCK OFFSETS


                           ;
                           ; START OF PURE CODE
                           ;
        000000                    .PSECT
```

```
ALOCB = ****** GX           H.CSP   000000        NF.END= 000200         S1.IBY= 000200      S5.BCC= 020000
ASSIGN  000026R        002  H.CUIC  000010        NF.EVT= 000002         S1.OBY= 000100      S5.DAO= 040000
A$$CHK= 000000              H.DSW   000046        NF.LLI= 000004         S1.PTH= 000010      S5.HPC= 000014
A$$CPS= 000000              H.DUIC  000012        NF.LV2= 000100         S1.RES= 010000      S5.HPD= 000020
A$$PRI= 000000              H.EFLM  000004        NF.SLI= 000400         S1.RNE= 000020      S5.ITI= 000100
A$$TRP= 000000              H.EFSV  000040        NF.SMC= 000020         S1.RNF= 020000      S5.OXF= 000040
A.LULU= 000002              H.FCS   000050        NF.SWP= 000010         S1.RSP= 000004      S5.RPO= 002000
A.LUNA= 000004              H.FORT  000052        N$$ACC= 000001         S1.RST= 000001      S5.SW1= 000001
A.LUNU= 000006              H.FPSA  000042        N$$BUF= 000001         S1.TNE= 040000      S5.TMM= 000002
BUFUMP= 172354              H.FPSL= 000062        N$$LDV= 000001         S1.TSY= 000040      S5.VER= 010000
CMODE = 140000              H.FPVA  000034        N$$MCP= 000001         S1.USI= 100000      S5.XOF= 000004
C$$CKP= 000000              H.GARD  000072        N$$MLL= 000001         S2.BEL= 020000      S5.XON= 000010
C$$ORE= 000400              H.HDLN  000002        N$$MOV= 000010         S2.BRQ= 000020      S6.EIO= 000400
C$$RSH= 177564              H.IPC   000016        N$$NCT= 000001         S2.CR = 002000      S6.RDI= 100000
DEACB = ****** GX           H.IPS   000014        N$$PEM= 000001         S2.CTO= 040000      S6.RLU= 001000
DV.CCL= 000002              H.ISP   000020        PDVID = ****** GX      S2.CTS= 100000      TPS   = 177564
DV.COM= 020000              H.LUN   000076        PIRQ  = 177772         S2.ELF= 001000      T$$KMG= 000000
DV.DIR= 000010              H.NLUN  000074        PMODE = 030000         S2.FLF= 000400      T$$MIN= 000000
DV.EXT= 000400              H.NML   000061        PR0   = 000000         S2.IRQ= 000200      UBMPR = 170200
DV.F11= 040000              H.ODVA  000022        PR1   = 000040         S2.OBF= 004000      UC.ALG= 000200
DV.ISP= 002000              H.ODVL  000024        PR2   = 000100         S2.DRQ= 000100      UC.ATT= 000010
DV.MBC= 000400              H.OVLY  000054        PR3   = 000140         S2.PCU= 010000      UC.KIL= 000004
DV.MNT= 100000              H.PFVA  000032        PR4   = 000200         S2.RCU= 000001      UC.LGH= 000003
DV.MSD= 000100              H.RCVA  000036        PR5   = 000240         S2.SRQ= 000040      UC.NPR= 000100
DV.OSP= 004000              H.RRVA  000062        PR6   = 000300         S2.WAL= 000010      UC.PWF= 000020
DV.PSE= 010000              H.SPRI  000060        PR7   = 000340         S2.WRA= 000006      UC.QUE= 000040
DV.REC= 000001              H.TKVA  000026        PS    = 177776         S2.WRB= 000002      UD.UNS= 000000
DV.SDI= 000020              H.TKVL  000030        P$$P45= 000000         S3.ACR= 000001      UD.160= 000004
DV.SQD= 000040              H.VEXT  000056        P$$WRD= 000000         S3.CTC= 000004      UD.200= 000001
DV.SWL= 001000              H.WND   000044        Q$$OPT= 000010         S3.FDX= 000200      UD.556= 000002
DV.TTY= 000004              H.X25   000064        R$R0  = 000002         S3.ICE= 001000      UD.625= 000005
DV.UMD= 000200              I$$RAR= 000000        R$R1  = 000004         S3.MHE= 000400      UD.8K = 000006
D$$BUG= 177514              I$$RDN= 000000        R$R2  = 000006         S3.NEC= 000020      UD.800= 000003
D$$ISK= 000000              KISAR0= 172340        R$R3  = 000010         S3.PPT= 020000      UISAR0= 177640
D$$L11= 000001              KISAR6= 172354        R$R4  = 000012         S3.PTH= 004000      UISAR1= 177642
D$$YNC= 000000              K$$CNT= 177546        R$R5  = 000014         S3.RAL= 000010      UM.CLI= 000036
D$$YNM= 000000              K$$CSR= 177546        R$$DER= 000000         S3.RES= 010000      UM.DSB= 000200
D.OSP   000012              K$$LDC= 000000        R$$K11= 000001         S3.RUB= 040000      UM.NBR= 000400
D.LNK   000000              K$$TPS= 000074        R$$SND= 000000         S3.TAB= 000002      UM.OVR= 000001
D.MSK   000014              LD$LP = 000000        R$$11M= 000000         S3.TME= 002000      US.ABO= 000001
D.NAM   000004              LONACP  000000R  002  R.UNGC= 000015         S3.TSY= 000040      US.BSP= 000002
D.PCB   000034              L$$ASG= 000000        R.UNPC= 000014         S3.8BC= 000100      US.BSY= 000200
D.UCB   000002              L$$DRV= 000000        R.UNPN= 000006         S4.ABD= 100000      US.FOR= 000040
D.UCBL  000010              L$$P11= 000001        R.UNPR= 000012         S4.ANI= 000400      US.FRK= 000002
D.UNIT  000006              L$$11R= 000000        R.UNRM= 000022         S4.AVO= 001000      US.KPF= 000001
D.VCAN= 000002              MPAR  = 172100        R.UNRU= 000024         S4.BLK= 002000      US.LAB= 000004
D.VDEB= 177776              MPCSR = 177746        R.UNSM= 000016         S4.DEC= 004000      US.MDE= 000002
D.VINI= 000000              M$$CRB= 000124        R.UNSU= 000020         S4.DLO= 000100      US.MDM= 000020
D.VOUT= 000004              M$$CRX= 000000        R.UNTN= 000002         S4.EDT= 010000      US.MNT= 000100
D.VPWF= 000006              M$$FCS= 000000        SWR   = 177570         S4.HFF= 000020      US.OFL= 000001
E$$XPR= 000000              M$$MGE= 000000        S$$WRG= 000000         S4.HFL= 000007      US.PUB= 000004
F$$LVL= 000001              M$$NET= 000000        S$$YSZ= 007600         S4.HHT= 000040      US.PWF= 000010
G$$TPP= 000000              M$$OVR= 000000        S1.DEC= 001000         S4.HSY= 000200      US.RED= 000002
G$$TSS= 000000              NF$ACC= 100000        S1.DPR= 000400         S4.RGS= 020000      US.SHR= 000001
G$$TTK= 000000              NF$EVT= 040000        S1.DSI= 004000         S4.SFC= 040000      US.SPU= 000002
G$$WRD= 000000              NF.ACC= 000001        S1.ESC= 000002         S4.VFL= 000010      US.UMD= 000010
HEADR = ****** GX           NF.BCP= 000040        S1.IBF= 002000         S5.ABP= 100000      US.VV = 000001
```

```
    274                                                                ; UNTIL NEXT BASE LEVEL          ;[MP01]
    275                                         CALL     $NXHCK        ; INIT DLX IF NECESSARY          ;[MP01]
    276
    277                                         .ENDC                                                   ;[MP01]
    278
    279 000202                                  CALL     $NITIM        ;GO START TIMERS
    280 000206  000241                          CLC                    ;GOOD FINISH
    281 000210                         40$:     RETURN
    282
    283
    284
```

```
NTICSR  MACRO V05.03b Saturday 29-Jun-85 03:30  Page 7-1
CSR CHECK

    153 000146  103325                          BCC     10$                     ;BR IF NO, CHECK NEXT
    154                                 ;
    155                                 ; FOUND A BAD CSR
    156                                 ;
    157 000150  016700  000016'                 MOV     CSRSL1,R0               ;PICK UP SLT ADDRESS
    158
    159                                         .IF NDF S$$BAS
    160
    161 000154                                  SAVRG   <R1>                    ;SAVE R1
    162 000156  116001  000002                  MOVB    L.DDM(R0),R1            ;GET DDM PDV INDEX
    163 000162  067701  000000G                 ADD     @PDVTA,R1               ;INDEX INTO PDV VECTOR TABLE
    164 000166  011101                          MOV     (R1),R1                 ;GET PDV ADDRESS
    165 000170  016101  000004                  MOV     Z.NAM(R1),R1            ;GET DDM NAME
    166
    167                                         .ENDC   ;NDF S$$BAS
    168
    169                                         .IFDF   R$$11S
    170
    171 000174  042760  042000  000000          BIC     #LF.RDY!LF.ENA,L.FLG(R0);INDICATE LINE IS UNUSABLE
    172
    173                                         .IFF    ;DF R$$11S
    174
    175                                         MOV     #LX.LIN,$NTLFC+LR.STS   ;SET "CLEAR LINE" FUNCTION CODE
    176                                         MOVB    L.CTL(R0),LR.CTL+$NTLFC ;SET CONTROLLER NUMBER
    177                                         MOVB    L.UNT(R0),LR.UNT+$NTLFC ;SET UNIT NUMBER
    178                                         MOV     R1,LR.LIN+$NTLFC        ;GET DDM NAME (RAD50)
    179                                         CALL    $NINTL                  ;SEND REQUEST TO NTL
    180
    181                                         .IFTF   ;DF R$$11S
    182
    183                                                 .IF NDF S$$BAS
    184
    185 000202  005067  000002G                 CLR     $NIERA+2                ;MAKE A NULL
    186 000206  116067  000012  000002G         MOVB    L.CTL(R0),$NIERA+2      ;SET CONTROLLER NUMBER
    187 000214  116000  000002                  MOVB    L.DDM(R0),R0            ;GET DDM PDV INDEX (SIGN BIT NEVER ON)
    188 000220  010167  000000G                 MOV     R1,$NIERA               ;SET RAD50 DEVICE/PDV NAME
    189 000224                                  EMSG$   20                      ;DEVICE "DDM-NUM" OFFLINE
    190 000232                                  RESRG   <R1>                    ;RESTORE R1
    191
    192                                                 .ENDC   ;NDF S$$BAS
    193
    194                                         .ENDC   ;DF R$$11S
    195 000234  000672                          BR      10$                     ;CHECK REST ANYWAY
    196
    197 000236  000241                  50$:    CLC                             ;ALWAYS RETURN SUCCESS
    198 000240                                  RESRG   <R2,R1,R0>              ;RESTORE REGISTERS
    199 000246                          60$:    RETURN
    200
    201                                 ;
    202                                 ; ILLEGAL ADDRESS TRAP ROUTINE, EXPECTS A ZERO ENTRY ON TOP OF USER STACK
    203                                 ;
    204 000250  005266  000004          CSRTRP: INC     4(SP)                   ;SET TOP OF USER STACK TO A "ONE"
    205 000254  000002                          RTI                             ;RETURN WITH "BAD CSR" INDICATION
    206
    207
    208         000001                          .END
```

```
NTICSR  MACRO V05.03b Saturday 29-Jun-85 03:30  Page 7-2
Symbol table
```

```
                                                                        E  5
NTIDLI11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30       PAGE 1
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE                REFERENCES

AREA     = 176000             #6-69         7-126        7-130        7-150        7-154
BUFUMP   = 172354             #6-62
CMODE    = 140000             #6-62
D$HOST     000022               7-153       *7-157
D$LNAM     000006              *7-139       *7-140       *7-141
D$LNUM     000014               7-129       *7-132
I$$AS    = ******               6-62
KISAR0   = 172340             #6-62
KISAR6   = 172354             #6-62         *7-117
MPAR     = 172100             #6-62
MPCSR    = 177746             #6-62
M$$MGE   = 000000               6-62
PIRQ     = 177772             #6-62
PMODE    = 030000             #6-62
PR0      = 000000             #6-62
PR1      = 000040             #6-62
PR2      = 000100             #6-62
PR3      = 000140             #6-62
PR4      = 000200             #6-62
PR5      = 000240             #6-62
PR6      = 000300             #6-62
PR7      = 000340             #6-62
PS       = 177776             #6-62
R$$MPL   = ******               6-62         6-62         6-62
R$$11D   = ******               6-62
R$$11S   = 000001             #4-2
SWR      = 177570             #6-62
TPS      = 177564             #6-62
UBMPR    = 170200             #6-62
UISAR0   = 177640             #6-62
UISAR1   = 177642             #6-62
$DECPT   = ******   GX          7-124
$NIDLI     000000 RG          #7-103
$SYSIZ   = ******   GX          7-103
$TTNS    = ******   GX          7-161
.BASEB   = 140000             #6-62         7-118
```

```
                                                                        F  5
NTIDLI11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30       PAGE 2
```

```
                                                                  E 6
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:31  Page 10-1
Symbol table

ALOCB = ****** GX       CM.CIR= 000002        C.FNC   000010        D$MLL   000040        FC.MLD= 000026
AREAMK= 176000          CM.FMT= 100000        C.LIN   000006        D$MNOD  000041        FC.PCT= 000030
ASSIGN  000036R     002 CM.HRD= 000002        C.LNK   000000        D$NA    000062        FC.PWR= 000022
AT$ACL= 000100          CM.LIN= 000000        C.MOD   000011        D$NBEA  000056        FC.RCE= 000002
AT$ADN= 000040          CM.LOO= 000001        C.NSP   000004        D$NBRA  000054        FC.RCP= 000014
AT$AUP= 000020          CM.XLO= 000004        C.PRO   000042        D$NEND= 000054        FC.TIM= 000010
AT$CYC= 000004          CP.DCF= 000040        C.RSV   000002        D$NLN   000030        FC.XCP= 000012
AT$LV1= 000002          CP.HDL= 000007        C.STA   000007        D$NN    000060        FC.XME= 000000
AT$LV2= 000001          CP.PS = 177400        C.STS   000012        D$QUIT  000043        FNDPDV= ****** GX
AT$NEX= 000200          CP.PSI= 000200        C.URM   177776        D$RETF  000050        FS.AST= 000000
AT$UP = 000010          CP.XCF= 000100        C.XACP  000004        D$RNN   000002        FS.CIB= 002000
A$CIR   000003          CP.2FR= 000030        C.XID   000035        D$RTMR  000076        FS.CRA= 001000
A$LEN   000004          CS.ABO= 000100        C.XLEN  000044        D$SEG   000036        FS.DIS= 013000
A$NID   000000          CS.BRO= 000002        C.XPLI  000040        D$SER   000032        FS.DVC= 001000
A$TM    000003          CS.BUF= 000200        C.XPT   000034        D$SQRL  000052        FS.ENB= 012000
A$TMI   000002          CS.CES= 000002        C.XSVC  000042        D$$BUG= 177514        FS.EXI= 001000
A$TSZ   000000          CS.CHN= 000010        C.XTC   000037        D$$ISK= 000000        FS.GET= 006000
A$TYP   000002          CS.CMP= 000200        C.X25   000036        D$$L11= 000001        FS.HLT= 000000
A$$CHK= 000000          CS.DCR= 000400        DECPT = ****** GX     D$$YNC= 000000        FS.INI= 000000
A$$CPS= 000000          CS.DEF= 000004        DV.CCL= 000002        D$$YNM= 000000        FS.KIL= 000000
A$$PRI= 000000          CS.DEV= 000002        DV.COM= 020000        D.DSP   000012        FS.LCL= 100000
A$$TRP= 000000          CS.DIS= 000040        DV.DIR= 000010        D.LNK   000000        FS.LTM= 001000
A.LULU= 000002          CS.ENA= 000001        DV.EXT= 000400        D.MSK   000014        FS.MNT= 004000
A.LUNA= 000004          CS.ENB= 000020        DV.F11= 040000        D.NAM   000004        FS.MSN= 014000
A.LUNU= 000006          CS.ERR= 100000        DV.ISP= 002000        D.PCB   000034        FS.REA= 001000
BUFUMP= 172354          CS.FTL= 001000        DV.MBC= 000400        D.UCB   000002        FS.RET= 000000
CB.CCB= 000002          CS.HCR= 000001        DV.MNT= 100000        D.UCBL  000010        FS.REZ= 003000
CB.DDM= 000040          CS.HFE= 002000        DV.MSD= 000100        D.UNIT  000006        FS.RLB= 002000
CB.DLC= 000020          CS.LST= 040000        DV.OSP= 004000        D.VCAN= 000002        FS.RNG= 011000
CB.RDB= 000004          CS.MTL= 004000        DV.PSE= 010000        D.VDEB= 177776        FS.RST= 000000
CB.SDB= 000010          CS.RNG= 000010        DV.REC= 000001        D.VINI= 000000        FS.RTN= 001000
CB.SLI= 000100          CS.ROV= 000004        DV.SDI= 000020        D.VOUT= 000004        FS.SET= 005000
CB.XLB= 000001          CS.RSN= 010000        DV.SQD= 000040        D.VPWF= 000006        FS.SFC= 005000
CC.LLC= 000200          CS.SHU= 000001        DV.SWL= 001000        E$NBR   000014        FS.SFR= 006000
CE.ABO= 100362          CS.SID= 000002        DV.TTY= 000004        E$NBS   000020        FS.SFS= 004000
CE.DAO= 100346          CS.STR= 000004        DV.UMD= 000200        E$NCR   000034        FS.SPW= 040000
CE.DIS= 100366          CS.SUC= 000001        D$AMXC  000072        E$NCS   000036        FS.STM= 000000
CE.ERR= 100370          CS.TMO= 020000        D$AMXH  000074        E$NIC   000044        FS.STP= 002000
CE.ILN= 100350          CS.XUR= 000004        D$ANN   000000        E$NLEN  000050        FS.STR= 001000
CE.LTO= 100356          C$LEN   000012        D$BRPR  000102        E$NLLA  000012        FS.TRM= 003000
CE.MOP= 100372          C$NID   000000        D$BRTM  000100        E$NLNK  000000        FS.WLB= 001000
CE.NTE= 100361          C$NXH   000002        D$DELF  000045        E$NML   000040        FS.XKL= 002000
CE.RTE= 100376          C$TIM   000010        D$DELW  000046        E$NMR   000024        FS.XOF= 010000
CE.SRC= 100364          C$$CKP= 000000        D$END = 000104        E$NMS   000030        FS.XON= 007000
CE.STP= 100352          C$$ORE= 000400        D$FNB   000034        E$NNOD  000002        FS.ZER= 002000
CE.TME= 100354          C$$RSH= 177564        D$HIOR  000024        E$NRT   000042        F$$LVL= 000001
CE.TMO= 100374          C.ADD   000034        D$HOST  000022        E$NRTP  000005        G$$TPP= 000000
CE.UNS= 100344          C.BID   000003        D$INAC  000044        E$NSEG  000010        G$$TSS= 000000
CF.CHN= 000001          C.BUF   000014        D$INCT  000042        E$NTIM  000046        G$$TTK= 000000
CF.EOM= 000004          C.BUF1  000014        D$IPL   000051        E$NUSE  000004        G$$WRD= 000000
CF.HDR= 000020          C.BUF2  000024        D$LID   000020        E$STRT  000006        HEADR = ****** GX
CF.LB = 100000          C.CNT   000020        D$LNAM  000006        E$$XPR= 000000        H.CSP   000000
CF.LIN= 000002          C.CNT1  000020        D$LNUM  000014        FC.CCP= 000020        H.CUIC  000010
CF.SOM= 000010          C.CNT2  000030        D$LST   000047        FC.CTL= 000006        H.DSW   000046
CF.SYN= 000040          C.FLG   000022        D$MAXC  000064        FC.KCP= 000016        H.DUIC  000012
CF.TRN= 000100          C.FLG1  000022        D$MAXH  000066        FC.KIL= 000004        H.EFLM  000004
CMODE = 140000          C.FLG2  000032        D$MAXV  000070        FC.MAN= 000024        H.EFSV  000040
```

```
                                                                  F 6
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:31  Page 10-2
```

```
                                20$:
                                        .ENDC

000164  116503  000014                  MOVB    L.NSTA(R5),R3           ;GET # OF STATIONS ON THIS LINE
000170  001516                          BEQ     50$                     ;IF EQ, NOT A MULTI-DROP LINE
000172  010246                          MOV     R2,-(SP)                ;SAVE REVERSE MAPPING TABLE POINTER
000174  011202                          MOV     (R2),R2                 ;GET POINTER TO REVERSE MAPPING TABLE/2
000176  006302                          ASL     R2                      ;GET REAL POINTER
000200  005046                          CLR     -(SP)                   ;INITIALISE BIT TO SET IN LINE FLAGS
000202  010546                          MOV     R5,-(SP)                ;SAVE SLT POINTER
000204  010346                          MOV     R3,-(SP)                ;SAVE COUNT OF STATIONS
000206  116567  000012  177567          MOVB    L.CTL(R5),CTLNUM        ;SAVE CONTROLLER NUMBER
000214  105067  177562                  CLRB    STANUM                  ;START WITH STATION NUMBER ZERO
000220  062705  000022                  ADD     #L.MPF,R5               ;POINT TO MULTI-DROP FLAGS TABLE
000224  132715  000100          30$:    BITB    #SF.ENA,(R5)            ;SHOULD THIS STATION BE ENABLED?
000230  001461                          BEQ     40$                     ;NO

000232  011246                          MOV     (R2),-(SP)              ;GET REVERSE MAPPING VALUE
000234  000316                          SWAB    (SP)                    ;GET OWNER LLC INDEX
000236  005001                          CLR     R1                      ; ...
000240  111601                          MOVB    (SP),R1                 ; ...
000242  005726                          TST     (SP)+                   ;CLEAN UP STACK
000244  067701  000000G                 ADD     @PDVTA,R1               ;GET OWNER LLC PDV ADDRESS
000250  011101                          MOV     (R1),R1                 ; ...
000252  022761  114224  000004          CMP     #^RXPT,Z.NAM(R1)        ;OWNER XPT ?
000260  001030                          BNE     36$                     ;IF NO - BRANCH

000262  016701  000000G                 MOV     $DBECL,R1               ;COPY ECL DATA BASE ADDRESS
000266  132761  000040  000005          BITB    #NF$MOU,N$FLG(R1)       ;IS NETWORK ON ?
000274  001001                          BNE     31$                     ;IF YES - BRANCH
000276  000436                          BR      40$                     ;IF NO - DO     ENABLE CIRCUIT

000300  017701  000000G         31$:    MOV     @DECPT,R1               ;GET DECNET HOME BLOCK ADDRESS
000304  005761  000060                  TST     D$NN(R1)                ;ROUTING NODE ?
000310  001014                          BNE     36$                     ;IF YES - BRANCH
000312  005767  177466                  TST     ACTCIR                  ;IF NO - ANY OTHER LINES ENABLED ?
000316  001407                          BEQ     35$                     ;IF NO - BRANCH
000320  142765  000100  000000          BICB    #SF.ENA,S.FLG(R5)       ;IF YES - DISABLE LINE
000326                                  EMSG$   12                      ;PRINT MSG
000334  000417                          BR      40$                     ;CONTINUE

000336  005267  177442          35$:    INC     ACTCIR                  ;COUNT ANOTHER XPT CIRCUIT ENABLED
000342  010567  177432          36$:    MOV     R5,STAADD               ;SAVE STATION TABLE ADDRESS
000346                                  CALL    ENALIN                  ;ENABLE THE LINE
000352  103410                          BCS     40$                     ;BR IF LINE NOT OWNED BY SPECIFIED LLC
000354  012766  100000  000004          MOV     #LF.ACT,4(SP)           ;WE WILL MAKE THE LINE ACTIVE
000362  152715  000200                  BISB    #SF.ACT,(R5)            ;MARK THE STATION ACTIVE
000366  112765  000000  000002          MOVB    #LN.ON,S.NMST(R5)       ;MARK THE MANAGEMENT STATE AS 'ON'
000374  022525                  40$:    CMP     (R5)+,(R5)+             ;MOVE ON TO NEXT STATION
000376  062702  000002                  ADD     #2,R2                   ;MOVE ON TO NEXT LINE
000402  105267  177374                  INCB    STANUM                  ;INCREMENT STATION NUMBER
000406  005316                          DEC     (SP)                    ;REDUCE COUNT OF STATIONS TO GO
000410  001305                          BNE     30$                     ;LOOP TILL ALL CHECKED
000412  005726                          TST     (SP)+                   ;REMOVE COUNT FROM STACK
000414  012605                          MOV     (SP)+,R5                ;RECOVER SLT ADDRESS
000416  052665  000000                  BIS     (SP)+,L.FLG(R5)         ;POSSIBLY MARK THE LINE ACTIVE
000422  012602                          MOV     (SP)+,R2                ;RECOVER REVERSE MAPPING TABLE ADDRESS
```

```
**FILE**ID**NTINIT
```

```
NN      NN   TTTTTTTTTT     IIIIII     NN      NN     IIIIII     TTTTTTTTTT
NN      NN   TTTTTTTTTT     IIIIII     NN      NN     IIIIII     TTTTTTTTTT
NN      NN       TT           II       NN      NN       II           TT
NN      NN       TT           II       NN      NN       II           TT
NNNN    NN       TT           II       NNNN    NN       II           TT
NNNN    NN       TT           II       NNNN    NN       II           TT
NN  NN  NN       TT           II       NN  NN  NN       II           TT
NN  NN  NN       TT           II       NN  NN  NN       II           TT
NN    NNNN       TT           II       NN    NNNN       II           TT
NN    NNNN       TT           II       NN    NNNN       II           TT
NN      NN       TT           II       NN      NN       II           TT          ....
NN      NN       TT           II       NN      NN       II           TT          ....
NN      NN       TT         IIIIII     NN      NN     IIIIII         TT          ....
NN      NN       TT         IIIIII     NN      NN     IIIIII         TT          ....
```

```
    11           11          SSSSSSSS
    11           11          SSSSSSSS
  1111         1111        SS
  1111         1111        SS
    11           11        SS
    11           11        SS
    11           11          SSSSSS
    11           11          SSSSSS
    11           11                SS
    11           11                SS
    11           11                SS
    11           11                SS
  111111       111111      SSSSSSSS
  111111       111111      SSSSSSSS
```

```

                .SBTTL  VECSET - SET UP PROCESS VECTOR TABLE
        ;+
        ;
        ; VECSET - SET UP PROCESS VECTOR TABLE
        ;
        ; INPUTS:
        ;       .CXVEC - LIST OF VECTOR BLOCKS
        ;
        ; OUTPUTS:
        ;       PROCESS VECTOR TABLES TRANSLATED
        ;
        ;-
                .IF DF  R$$MPL

        VECSET: SWSTK$  20$                     ; ENTER SYSTEM STATE
                SAVMAP                          ; SAVE MAPPING
                MOV     @NTLPT,@KSAR6           ; MAP TO NTL HOME BLOCK
                MOV     140000+.CXVEC,R5        ; GET VECTOR BLOCK ADDRESS
                BEQ     15$                     ; IF NONE- BRANCH
                MOV     R5,VECLHD               ; SAVE LISTHEAD FOR DEALLOCATION
                CLR     140000+.CXVEC           ; INIT VECTOR BLOCK LISTHEAD

        5$:     MOV     R5,-(SP)                ; SAVE UNMAPPED POOL.. ADDRESS
                CALL    @CEACC                  ; ...
                MOV     (SP)+,R5                ; GET MAPPED VIRTUAL ADDRESS
                MOV     V.ADD(R5),R3            ; COPY TABLE ADDRESS OF PROCESS
                BIC     #160000,R3              ; BIAS FOR APR6
                BIS     #140000,R3              ; ...
                MOV     V.SIZ(R5),R2            ; COPY TABLE LENGTH
                MOV     V.PDV(R5),R4            ; GET PDV ADDRESS OR LIBRARY TABLE BIAS
                SAVMAP                          ; SAVE MAPPING TO VECTOR BLOCKS

                BIT     #VF.LIB,V.FLG(R5)       ; LIBRARY VECTORED BLOCK ?
                BEQ     6$                      ; IF NO - BRANCH
                MOV     R4,@KSAR6               ; MAP TO LIBRARY
                BR      9$                      ; CONTINUE

        6$:     MOV     Z.DSP(R4),@KSAR6        ; MAP TO PROCESS

        9$:     MOV     @#112,R1                ; COPY BIAS OF EXEC ROUTINE WHICH
                                                ; CALLS THE VECTOR TRANSLATE ROUTINE
                MOV     (R1),R1                 ; GETTING TO ACTUAL BIAS
                MOV     (R1),-(SP)              ; PUT BIAS ON STACK FOR $MPPRO
                MOV     #120000,-(SP)           ; MAP THRU APR5
                CALL    @MPPRO                  ; CALL DIRECTIVE COMMON ROUTINE THE LONG WAY
                RESMAP                          ; RESTORE MAPPING TO VECTOR BLOCKS

                MOV     (R5),R5                 ; GET NEXT BLOCK
                BNE     5$                      ; IF MORE - BRANCH

                MOV     VECLHD,R5               ; DEALLOCATE BLOCKS (AUX VECTOR TABLE IS
                                                ; NOW SET UP)
        10$:    MOV     R5,-(SP)                ; COPY UNMAPPED ADDRESS
                MOV     (SP),-(SP)              ; SAVE UNMAPPED ADDRESS
                CALL    @CEACC                  ; MAP TO BLOCK
                MOV     (SP)+,R5                ; COPY VIRTUAL ADDRESS OF BLOCK
```

```
                                                                                 E 10
NTINIT  MACRO V05.03b Thursday 05-Sep-85 11:09  Page 18-3
Symbol table

$DEACB= ****** GX        $NERDN  000526RG        $NIXPT  000002RG     002 $T1SCL= ****** GX        .CXLB1  000030 G
$DECPT= ****** GX        $NER50  000504RG        $NTLPT= ****** GX        $T100C= ****** GX        .CXLB2  000044 G
$DEVHD= ****** GX        $NIAST  000120R      002 $PDVID= ****** GX        $T100Q= ****** GX        .CXLEN  000064 G
$ERR03  000246R      002 $NICEX= ****** GX       $PDVNM= ****** GX        $UMRPT= ****** GX        .CXPCB  000006 G
$ERR04  000300R      002 $NIECL  000000RG     002 $PDVTA= ****** GX        $$$    = 000062           .CXSYM  000010 G
$ERR12  000336RG     002 $NIERA  000010RG     002 $PSIPT= ****** GX        $$$ARG= 000002           .CXUNL  000024 G
$FMASK= ****** GX        $NIERR  000332RG        $PUMR  = ****** GX        $$$OST= 000016           .CXVEC  000026 G
$HEADR= ****** GX        $NIEVC  000102RG     002 $QASTT= ****** GX        $$$T1  = 000006           .CXVSM  000016 G
$LDPWF= ****** GX        $NIFNC= 000032RG     002 $SLTMA= ****** GX        $$$$   = 000001           .NTGCD  000000
$LGSTT= ****** GX        $NILON= ****** GX       $SLTNM= ****** GX        .BASEB= 140000           .NTPCB  000000 G
$LLCAL= ****** GX        $NIMRK  000070R      002 $SRSTD= ****** GX        .CXALL  000062 G         .NTUMR  000002 G
$LLCRS= ****** GX        $NIPRM  000122R      002 $TKPS  = ****** GX        .CXCSR  000004 G         .SBPCB  000066 G
$LLCTA= ****** GX        $NIRDB  000032RG     002 $TKTCB= ****** GX        .CXLBR  000060 G         .$$$$.= 000034
$NERDD  000550RG         $NIRTN  000026RG     002 $TK100= ****** GX

. ABS.  177776    000    (RW,I,GBL,ABS,OVR)
        000576    001    (RW,I,LCL,REL,CON)
DATA    000532    002    (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 116
Work  file writes: 107
Size of work file: 25359 Words  ( 100 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:01:36.51
DB2:NTINIT11S.T34,[132,134]NTINIT11S/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]T34,NTINIT
```

```
                                                                                 F 10
NTINIT11S  CREATED BY  MACRO  ON 5-SEP-85 AT 11:10       PAGE 1
```

```
    121                                                .SBTTL  T1SIN - INSERT 1 SECOND TIMER FORK BLOCK
    122                                                .SBTTL  T50IN - INSERT 50 MSEC TIMER FORK BLOCK
    123                                         ;+
    124                                         ; *** - T1SIN - INSERT 1 SECOND TIMER FORK BLOCK
    125                                         ; *** - T50IN - INSERT 50 MSEC TIMER FORK BLOCK
    126                                         ;
    127                                         ; THIS ROUTINE IS USED TO INSERT ENTRIES ON THE CLOCK QUEUE, WITHOUT CHANGING
    128                                         ; THE PREDEFINED APR5 VALUE STORED IN THE CLOCK BLOCK. THIS ROUTINE IS ONLY
    129                                         ; NEEDED ON MAPPED SYSTEMS.
    130                                         ;
    131                                         ; THIS ROUTINE SHOULD BE CALLED AT SYSTEM STATE - TO INHIBIT TASK SWITCHING
    132                                         ;
    133                                         ; INPUTS:
    134                                         ;       NONE.
    135                                         ;
    136                                         ; OUTPUTS:
    137                                         ;       SPECIFIED CLOCK BLOCK IN INSERT IN CLOCK QUEUE
    138                                         ;
    139                                         ;-
    140                                                 .ENABL  LSB
    141 000116                                  T1SIN:
    142 000116  016700  000000G                         MOV     T1SCL,R0                ;* GET ADDRESS OF CLOCK QUEUE EN
    143 000122  017702  000000G                         MOV     @TKPS,R2                ;* GET # OF CLOCK TICKS PER SECO
    144 000126  000407                                  BR      10$                     ;* JOIN COMMON CODE
    145 000130                                  T50IN:
    146 000130  005777  000000G                         TST     @T100Q                  ;* ANYTHING IN THE 50 MSEC TIMER QUEUE ?
    147 000134  001411                                  BEQ     20$                     ;* IF EQ, NO .. DON'T START THE TIMER
    148 000136  016700  000000G                         MOV     T100C,R0                ;* ELSE, GET ADDRESS OF CLOCK QUEUE ENTRY
    149 000142  017702  000000G                         MOV     @TK100,R2               ;* GET # OF CLOCK TICKS IN 50 MSEC
    150 000146  005001                          10$:    CLR     R1                      ;* NO HIGH ORDER # OF TICKS
    151 000150  012704  000006                          MOV     #6,R4                   ;* SET SYSTEM REQUEST INDICATOR
    152 000154                                          CALLR   CLINS                   ;* INSERT CLOCK BLOCK IN QUEUE
    153 000160                                  20$:    RETURN
    154                                                 .DSABL  LSB
```

```
                                                                 E 12
NTIUMR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:33     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE              REFERENCES

M.BFVH     000011            7-87
M.BFVL     000012            7-86
M.UMRA     000002            7-88         7-90
M.UMRN     000004            7-89
R$$11S   = 000001           #4-2
$NIUMR     000000 RG        #7-84
```

```
                                                                 F 12
NTIUMR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:33     PAGE 2
```

```
**FILE**ID**CFEDEF
```

```
  CCCCCCCC  FFFFFFFFFF  EEEEEEEEEE  DDDDDDDD    EEEEEEEEEE  FFFFFFFFFF
  CCCCCCCC  FFFFFFFFFF  EEEEEEEEEE  DDDDDDDD    EEEEEEEEEE  FFFFFFFFFF
CC          FF          EE          DD      DD  EE          FF
CC          FF          EE          DD      DD  EE          FF
CC          FF          EE          DD      DD  EE          FF
CC          FF          EE          DD      DD  EE          FF
CC          FFFFFFFF    EEEEEEEE    DD      DD  EEEEEEEE    FFFFFFFF
CC          FFFFFFFF    EEEEEEEE    DD      DD  EEEEEEEE    FFFFFFFF
CC          FF          EE          DD      DD  EE          FF
CC          FF          EE          DD      DD  EE          FF
CC          FF          EE          DD      DD  EE          FF              ....
CC          FF          EE          DD      DD  EE          FF              ....
  CCCCCCCC  FF          EEEEEEEEEE  DDDDDDDD    EEEEEEEEEE  FF              ....
  CCCCCCCC  FF          EEEEEEEEEE  DDDDDDDD    EEEEEEEEEE  FF              ....


LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

```
                                                 .SBTTL  SETSYS - PROCESS ''SET SYSTEM'' REQUEST
                                         ;+
                                         ; *** - SETSYS - PROCESS ''SET SYSTEM'' REQUEST
                                         ;
                                         ; INPUT:
                                         ;       $OPRND = OPERAND WORD
                                         ;
                                         ; OUTPUT:
                                         ;       CONTROL IS DISPATCHED TO THE PROPER ROUTINE
                                         ;
                                         ;-

001174                                   SETSYS:
001174  032767  000000G 000000G                  BIT     #OS.PRT,$OPRND   ; CHANGE THE PARTITION DEFINITION ?
001202  001402                                   BEQ     10$              ; IF EQ, NO
001204  004767  000000G                          CALL    $SEPAR           ; CHANGE THE PARTITION DEFINITION
001210  032767  000000G 000000G  10$:            BIT     #OS.BUF,$OPRND   ; CHANGE THE BUFFER DEFINITION ?
001216  001402                                   BEQ     20$              ; IF EQ, NO .. LEAVE
001220  004767  000000G                          CAL     $SEBUF           ; CHANGE THE BUFFER DEFINITION
001224  000207                   20$:            RETURN

```

E 15

```
**FILE**ID**CFEHLP




  CCCCCCCC   FFFFFFFFFF   EEEEEEEEEE   HH      HH   LL           PPPPPPPP
  CCCCCCCC   FFFFFFFFFF   EEEEEEEEEE   HH      HH   LL           PPPPPPPP
CC           FF           EE           HH      HH   LL           PP      PP
CC           FF           EE           HH      HH   LL           PP      PP
CC           FF           EE           HH      HH   LL           PP      PP
CC           FF           EE           HH      HH   LL           PP      PP
CC           FFFFFFFF     EEEEEEEE     HHHHHHHHHH   LL           PPPPPPPP
CC           FFFFFFFF     EEEEEEEE     HHHHHHHHHH   LL           PPPPPPPP
CC           FF           EE           HH      HH   LL           PP
CC           FF           EE           HH      HH   LL           PP
CC           FF           EE           HH      HH   LL           PP                ....
CC           FF           EE           HH      HH   LL           PP                ....
  CCCCCCCC   FF           EEEEEEEEEE   HH      HH   LLLLLLLLLL   PP                ....
  CCCCCCCC   FF           EEEEEEEEEE   HH      HH   LLLLLLLLLL   PP                ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

F 15

CFEHLP   HELP ROUTINE FOR CFE   MACRO V05.03b Saturday 29-Jun-85 04:55

```
DATA DEFINITIONS FOR FILE I/O

    89                                                .SBTTL  DATA DEFINITIONS FOR FILE I/O
    90                                        ;
    91                                        ; DEFINE THE FSR REGION
    92 000000                                         FSRSZ$  3                 ; DEFINE SIZE OF FSR REGION
    93                                                                          ; ALLOW FOR 3 FILES OPEN: GCL, INPUT, OUTPUT
    94
    95 000000                                         GCLDF$  LUN1,LUN1,<CFE>,$BUFF,BUFSIZ-1
    96                                                .GLOBL  CL.CMT,CL.EIF,CL.LCC,CL.OPN,CL.XIT,CL.RNE
    97 000000  000207                         $CLRTN: RETURN                    ; FUDGE LABEL FOR GCLDF MACRO
    98
    99 000002                                         FDOFF$  DEF$L             ; DEFINE FDB OFFSETS
   100                                                .GLOBL  F.NRBD,F.URBD
   101                                        ;
   102                                        ; DEFINE THE OUTPUT FILE FDB
   103                                        ;
   104 000002                                 $OFDB:: FDBDF$
   105 000142                                         FDAT$A  R.VAR,FD.CR       ; VAR LENGTH FILE, NORMAL CARRIAGE CTL
   106 000142                                         FDOP$A  LUN4,,$DEFNB,FO.WRT ; OPEN THE FILE "SY:CETAB.MAC" FOR WRITE
   107                                        ;
   108                                        ; DEFINE THE INPUT FILE FDB
   109                                        ;
   110 000142                                 $IFDB:: FDBDF$
   111 000302                                         FDAT$A  R.VAR,FD.CR       ; VAR LENGTH FILE, NORMAL CARRIAGE CTL
   112 000302                                         FDRC$A  ,$BUFF,BUFSIZ     ; POINT AT INPUT BUFFER
   113 000302                                         FDOP$A  LUN5,,$DEFNB,FO.UPD ; OPEN FILE "SY:CETAB.MAC" FOR UPDATE
   114
   115 000302                                 $DEFNB::NMBLK$  CETAB,MAC,,SY,0 ; DEFAULT NAME IS "SY:CETAB.MAC"
   116
   117 000340                                 $FIATR::                          ; ATTRIBUTE LIST FOR IO.RAT/IO.WAT QIO
   118 000340     377     004                         .BYTE   -1,4              ; READ OWNER UIC AND PROTECTION
   119 000342  000346' 000000                         .WORD   10$, 0
   120 000346                                 10$:    .BLKW   3                 ; BUFFER FOR ATTRIBUTES
   121
```

```
                                        .SBTTL  LOAD UMR'S

                                ;+
                                ; $NIUMR - INITIALIZE UMR'S ON AN 11/70 SYSTEM
                                ;
                                ; PLEASE NOTE THAT THIS CODE IS ALWAYS ASSEMBLED FOR AN 11/70 SYSTEM.
                                ; IT SHOULD NEVER BE CALLED IF THERE ARE NO UMR'S TO LOAD.
                                ;
                                ; INPUTS:
                                ;       R2 = UMR BLOCK ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       R2 = DESTROYED
                                ;-

000000  010346                  $NIUMR::MOV     R3,-(SP)                ;SAVE R3
000002  010446                          MOV     R4,-(SP)                ;SAVE R4
000004  016203  000012                  MOV     M.BFVL(R2),R3           ;GET PARTITION ADDRESS
000010  116204  000011                  MOVB    M.BFVH(R2),R4           ;...
000014  016246  000002                  MOV     M.UMRA(R2),-(SP)        ;GET FIRST UMR ADDRESS
000020  066216  000004                  ADD     M.UMRN(R2),(SP)         ;SET UPPER UMR LIMIT
000024  016202  000002                  MOV     M.UMRA(R2),R2           ;POINT AT FIRST UMR
000030  010322                  10$:    MOV     R3,(R2)+                ;LOAD THE ADDRESS TO MAP
000032  010422                          MOV     R4,(R2)+                ;  INTO A UMR
000034  062703  020000                  ADD     #20000,R3               ;SET ADDRESS TO NEXT 4K WORDS
000040  005504                          ADC     R4                      ;...
000042  020216                          CMP     R2,(SP)                 ;MORE UMR'S TO LOAD?
000044  103771                          BLO     10$                     ;BR IF NO, STILL BELOW UPPER LIMIT
000046  005726                          TST     (SP)+                   ;DUMP UPPER LIMIT
000050  012604                          MOV     (SP)+,R4                ;RECOVER R4
000052  012603                          MOV     (SP)+,R3                ;RECOVER R3
000054                                  RETURN
```

```
                                                                                              F  2
NTIX25 - NTINIT 'MODULE X25-SER MACRO V05.03b Saturday 29-Jun-85 03:29  Page 7-3
Symbol table

US.WCK= 000010       U.FNUM= 000040       U.TTYP  000046       U3.OPA= 100000       ZF.LMC= 000100
UU.ABO= 000400       U.GRP = 000102       U.TUX   000024       U3.PAR= 040000       ZF.MAN= 020000
UU.ATN= 000100       U.KCSR= 000032       U.UHVR= 000107       U3.UPC= 020000       ZF.MFL= 000010
UU.AVN= 000004       U.KCS6= 000034       U.UIC   000044       U4.CR = 000100       ZF.MTM= 000400
UU.GUS= 000010       U.LUIC  177774       U.UNFL= 000052       V$GENQ  000024       ZF.MUX= 000040
UU.IOS= 002000       U.MEDI= 000040       U.UNIT  000006       V$LABL  000004       ZF.PSE= 002000
UU.ONL= 000020       U.MLUN= 000050       U.UNSZ= 000114       V$LEN   000034       ZF.SLI= 010000
UU.RCT= 000002       U.MUP   177772       U.UNTI= 000060       V$MBOX  000030       ZF.TIM= 000200
UU.RDY= 000200       U.OWN   177776       U.USVR= 000106       V$PKSR  000020       ZF.X3P= 000000
UU.SER= 000001       U.RBNS= 000112       U.UTIL= 000036       V$TRCT  000000       ZS.ASN= 100000
UU.SIO= 001000       U.RCTC= 000113       U.VCB = 000034       V$TYPE  000002       ZS.BSY= 140000
UU.SPC= 000040       U.RCTS= 000110       U.VSER= 000120       V$VCHA  000003       Z.AVL   000014
U.ACB   000062       U.RED   000002       U.2MED= 000070       V$$CTR= 001000       Z.DAT   000016
U.ACP = 000032       U.RED2  000034       U2.AT.= 000020       W.BATT  000006       Z.DSP   000000
U.ADMA  000066       U.SCB   000020       U2.CRT= 002000       W.BFPD  000014       Z.FLG   000010
U.AFLG  000064       U.SHST= 000076       U2.DH1= 100000       W.BHVR  000004       Z.LEN = 000016
U.ATT   000022       U.SHUN= 000074       U2.DJ1= 040000       W.BLGH  000020       Z.LLN   000006
U.BPKT= 000044       U.SPC = 000036       U2.DZ1= 000100       W.BLPD  000016       Z.MAP   000020
U.BUF   000024       U.STS   000005       U2.ESC= 001000       W.BLVR  000002       Z.NAM   000004
U.CBF = 000032       U.ST2   000007       U2.HFF= 010000       W.BNPD  000015       Z.PCB   000012
U.CLI   177772       U.SUB = 000036       U2.HLD= 000040       W.BOFF  000012       Z.SCH   000007
U.CNT   000030       U.TCHP  000042       U2.LOG= 000400       W.BPCB  000000       $ERR40  000036R     002
U.COTQ  000030       U.TCVP  000043       U2.LWC= 000001       W.BSIZ  000010       $ERR41  000076R     002
U.CTCB  000026       U.TFLK  000040       U2.L3S= 000004       X$$DBT= 000000       $ERR42  000136R     002
U.CTL   000004       U.TFRQ  000036       U2.L8S= 010000       ZF.COU= 001000       $HEADR= ****** GX
U.CTYP  000052       U.TIXL  000060       U2.NEC= 004000       ZF.DDM= 000001       $NIERR= ****** GX
U.CW1   000010       U.TLPP  000053       U2.PRV= 000010       ZF.DIA= 004000       $NIMFE= ****** GX
U.CW2   000012       U.TMTI  000047       U2.RMT= 020000       ZF.DLC= 000002       $NIX25  000000RG
U.CW3   000014       U.TRCK= 000100       U2.R04= 100000       ZF.DVP= 100000       $$$OST= 000010
U.CW4   000016       U.TSTA  000026       U2.SCS= 000004       ZF.INI= 040000       $$$T1 = 000002
U.CYL = 000104       U.TST5  000054       U2.SLV= 000200       ZF.KMX= 000020       $$$T2 = 000032R     002
U.DCB   000000       U.TST6  000056       U2.VT5= 000002       ZF.LLC= 000004       .BASEB= 140000
U.FCDE= 000042       U.TTAB  000050       U2.7CH= 010000

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000222    001   (RW,I,LCL,REL,CON)
DATA    000206    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 16287 Words  ( 64 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:17.11
SY:NTIX25.V2,[132,134]NTIX25/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTIX25
```

```
                                                                                              G  2
NTIX25     CREATED BY  MACRO  ON 29-JUN-85 AT 03:29     PAGE 1
```

```
                        ;+
                        ; *** - INILIN - REINITIALIZE HRC
                        ;
                        ; THIS SUBROUTINE IS USED AS A GENERALIZED INTERFACE TO REQUEST HRC...
                        ; TO REINITIALIZE ITSELF AFTER THE I/O DATA BASE HAS BEEN CHANGED.
                        ;
                        ; INPUT:
                        ;       NONE.
                        ;
                        ; OUTPUT:
                        ;       HRC... INITIALIZED OR ERROR MESSAGE PRINTED
                        ;
                        ;-

                                .IF DF  R$$MPL
                        INILIN: CALL    $SAVAL                  ; SAVE REG'S
                                MOV     @SLTMA,R5               ; GET START OF SLT'S
                                MOV     @SLTNM,R4               ; GET NUMBER OF SLT'S

                        1$:     MOV     (R5)+,R3                ; GET NEXT SLT ADDRESS
                                BIT     #LF.RDY,L.FLG(R3)       ; IS AT LEAST ONE LINE LOADED ?
                                BNE     5$                      ; IF YES - INIT HRC...
                                DEC     R4                      ; COUNT ONE LESS SLT
                                BNE     1$                      ; IF MORE - BRANCH
                                BR      20$                     ; CONTINUE

                        5$:     ALUN$S  #3,#"RD,#0              ; ASSIGN LUN TO RDO:
                                BCS     20$                     ; IF CS, EXIT

                                QIOW$S  #IO.ATT,#3,#32.,,#STATUS ; ATTACH TO THE RDO:
                                BCS     20$                     ; IF CS, ERROR
                                QIOW$S  #IO.HRC,#3,#32.,,#STATUS,,<#M$INIT,#M$INIT>
                                                                ; REQUEST HRC... TO
                                                                ; REINITIALIZE ITSELF
                                ROR     R1                      ; SAVE ERROR STATUS
                                QIOW$S  #IO.DET,#3,#32.         ; DETACH FROM RECONFIGURATION DRIVER
                                ROL     R1                      ; RESTORE CARRY
                                BCS     10$                     ; IF CS, EXIT
                                TSTB    STATUS                  ; WAS THE QIO SUCCESSFUL ?
                                BPL     20$                     ; IF PL, YES
                        10$:    EMSG$R  11                      ; ELSE, INDICATE ERROR

                        20$:    RETURN
                                .ENDC


        000001                  .END
```

```
NTICSR  MACRO V05.03b Saturday 29-Jun-85 03:30  Page 7-2
Symbol table

A$$CHK= 000000            LF.BWT= 000007      L.DLM   000006      PS    = 177776      ZF.COU= 001000
A$$CPS= 000000            LF.ENA= 002000      L.DLS   000010      P$$P45= 000000      ZF.DDM= 000001
A$$PRI= 000000            LF.LPB= 001000      L.FLG   000000      P$$WRD= 000000      ZF.DIA= 004000
A$$TRP= 000000            LF.MDC= 000100      L.KRBA  000016      Q$$OPT= 000010      ZF.DLC= 000002
BUFUMP= 172354            LF.MFL= 004000      L.LEN = 000022      R$$DER= 000000      ZF.DVP= 100000
CMODE = 140000            LF.MTP= 000020      L.MPF   000022      R$$K11= 000001      ZF.INI= 040000
CSRDIS  000006R       002 LF.PAC= 000200      L.NMST  000020      R$$SND= 000000      ZF.KMX= 000020
CSRENA  000000R       002 LF.RDY= 040000      L.NSTA  000014      R$$11M= 000000      ZF.LLC= 000004
CSRSLT  000016R       002 LF.REA= 010000      L.OWNR  000021      R$$11S= 000001      ZF.LMC= 000100
CSRTAB  000014R       002 LF.SER= 000040      L.UNT   000013      SF.ACT= 000200      ZF.MAN= 020000
CSRTRP  000250R           LF.TIM= 000010      MPAR  = 172100      SF.ENA= 000100      ZF.MFL= 000010
C$$CKP= 000000            LF.UNL= 020000      MPCSR = 177746      SF.LPB= 000004      ZF.MTM= 000400
C$$ORE= 000400            LF.X2P= 000000      M$$CRB= 000124      SF.MFL= 000040      ZF.MUX= 000040
C$$RSH= 177564            LN.CLO= 000000      M$$CRX= 000000      SF.PAC= 000020      ZF.PSE= 002000
D$$BUG= 177514            LN.DUM= 000005      M$$FCS= 000000      SF.REA= 000010      ZF.SLI= 010000
D$$ISK= 000000            LN.LOA= 000004      M$$MGE= 000000      SF.SER= 000001      ZF.TIM= 000200
D$$L11= 000001            LN.LOO= 000003      M$$NET= 000000      SF.SVC= 000002      ZF.X3P= 000000
D$$YNC= 000000            LN.OAU= 000003      M$$OVR= 000000      SF.UNL= 000040      ZS.ASN= 100000
D$$YNM= 000000            LN.OFF= 000001      N$$ACC= 000001      SWR   = 177570      ZS.BSY= 140000
E$$XPR= 000000            LN.ON = 000000      N$$BUF= 000001      S$$WRG= 000000      Z.AVL   000014
F$$LVL= 000001            LN.OOP= 000004      N$$LDV= 000001      S$$YSZ= 007600      Z.DAT   000016
G$$TPP= 000000            LN.OPE= 000001      N$$MCP= 000001      S.COST  000001      Z.DSP   000000
G$$TSS= 000000            LN.REF= 000002      N$$MLL= 000001      S.FLG   000000      Z.FLG   000010
G$$TTK= 000000            LN.SER= 000002      N$$MOV= 000010      S.LEN   000004      Z.LEN = 000016
G$$WRD= 000000            LN.STA= 000017      N$$NCT= 000001      S.NMST  000002      Z.LLN   000006
I$$RAR= 000000            LN.SUB= 000360      N$$PEM= 000001      S.OWNR  000003      Z.MAP   000020
I$$RDN= 000000            LN.TRI= 000006      PDVTA = ****** GX   S.VTTA= 000002      Z.NAM   000004
KISAR0= 172340            L$$ASG= 000000      PIRQ  = 177772      S.VTTL= 000004      Z.PCB   000012
KISAR6= 172354            L$$DRV= 000000      PMODE = 030000      TPS   = 177564      Z.SCH   000007
KSAR6 = ****** GX         L$$P11= 000001      PR0   = 000000      T$$KMG= 000000      $ERR20  000020RG      002
K$$CNT= 177546            L$$11R= 000000      PR1   = 000040      T$$MIN= 000000      $NERDN= ****** GX
K$$CSR= 177546            L.COST  000015      PR2   = 000100      UBMPR = 170200      $NICSR  000000RG
K$$LDC= 000000            L.CTL   000012      PR3   = 000140      UISAR0= 177640      $NIERA= ****** GX
K$$TPS= 000074            L.CVA   177776      PR4   = 000200      UISAR1= 177642      $NIERR= ****** GX
LD$LP = 000000            L.DDM   000002      PR5   = 000240      V$$CTR= 001000      $$$OST= 000006
LF.ACT= 100000            L.DDS   000004      PR6   = 000300      X$$DBT= 000000      .BASEB= 140000
LF.BRO= 000400            L.DLC   000003      PR7   = 000340

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000256    001   (RW,I,LCL,REL,CON)
DATA    000052    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 14216 Words  ( 56 Pages)
Size of core pool: 15496 Words  ( 59 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:13.05
SY:NTICSR11S.V2,[132,134]NTICSR11S/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]V2,NTICSR
```

```
NTICSR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 1
```

F 5

```
NTIDLI11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 2
MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME        REFERENCES

CALL               7-116
DHBDF$            #6-60       6-61
NHWDF$            #6-60       6-62
RETURN             7-168
SOB                7-166
SWSTK$             7-116
```

G 5

```
**FILE**ID**NTILON
```

```
                                                                 F  6
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:31  Page 10-2
Symbol table

H.FCS   000050        NF.ACC= 000001        N$$ACC= 000001        PT$XAR= 000020        R.UNTN= 000002
H.FORT  000052        NF.BCP= 000040        N$$BUF= 000001        PX$BLK= 000040        SRSTD = ****** GX
H.FPSA  000042        NF.END= 000200        N$$LDV= 000001        PX$DLM= 000200        SWR   = 177570
H.FPSL= 000062        NF.EVT= 000002        N$$MCP= 000001        PX$SVC= 000100        S$$WRG= 000000
H.FPVA  000034        NF.LLI= 000004        N$$MLL= 000001        P$CHN   000004        S$$YSZ= 007600
H.GARD  000072        NF.LV2= 000100        N$$MOV= 000010        P$CNT   000005        S1.DEC= 001000
H.HDLN  000002        NF.SLI= 000400        N$$NCT= 000001        P$CTR   000034        S1.DPR= 000400
H.IPC   000016        NF.SMC= 000020        N$$PEM= 000001        P$DRCT  000015        S1.DSI= 004000
H.IPS   000014        NF.SWP= 000010        PDVTA = ****** GX     P$DRTR  000036        S1.ESC= 000002
H.ISP   000020        N$ACQ   000000        PF$BLK= 020000        P$FLG   000012        S1.IBF= 002000
H.LUN   000076        N$ACTL  000032        PF$CLC= 010000        P$FWD   000030        S1.IBY= 000200
H.NLUN  000074        N$ADJ1  000072        PF$DLM= 100000        P$ICCB  000046        S1.OBY= 000100
H.NML   000061        N$ADJ2  000074        PF$EIP= 000002        P$IPL   000014        S1.PTH= 000010
H.ODVA  000022        N$CACH  000062        PF$ENB= 000001        P$LCD   000002        S1.RES= 010000
H.ODVL  000024        N$CIR   000034        PF$FAI= 004000        P$LEN   000052        S1.RNE= 000020
H.OVLY  000054        N$CRC   000120        PF$FM1= 000100        P$LST   000000        S1.RNF= 020000
H.PFVA  000032        N$DLA   000020        PF$FM2= 000200        P$NRNI  000040        S1.RSP= 000004
H.RCVA  000036        N$DLY   000014        PF$OFF= 000000        P$OCCB  000050        S1.RST= 000001
H.RRVA  000062        N$ELEN  000054        PF$RM1= 000020        P$PFQ   000006        S1.TNE= 040000
H.SPRI  000060        N$ENC   000042        PF$RM2= 000040        P$PKSZ  000044        S1.TSY= 000040
H.TKVA  000026        N$ERRC  000022        PF$RVR= 000010        P$RMX1  000016        S1.USI= 100000
H.TKVL  000030        N$FLG   000005        PF$STA= 000004        P$RMX2  000020        S2.BEL= 020000
H.VEXT  000056        N$FNC   000006        PF$SVC= 040000        P$RPRI  000042        S2.BRQ= 000020
H.WND   000044        N$GENQ  000052        PF$UP = 000004        P$RTIM  000003        S2.CR = 002000
H.X25   000064        N$GTM   000015        PIRQ  = 177772        P$STA1  000022        S2.CTO= 040000
I$$RAR= 000000        N$HC1   000052        PMODE = 030000        P$STA2  000023        S2.CTS= 100000
I$$RDN= 000000        N$HC2   000056        PR$BED= 000200        P$TIM   000010        S2.ELF= 001000
KISAR0= 172340        N$HIGH  000033        PR$BEU= 000100        P$TSCT  000026        S2.FLF= 000400
KISAR6= 172354        N$LLT   000026        PR$BRD= 000040        P$TSIZ  000024        S2.IRQ= 000200
K$$CNT= 177546        N$LLTM  000024        PR$BRU= 000020        P$TYP   000001        S2.OBF= 004000
K$$CSR= 177546        N$LVC   000036        PR$DWN= 000002        P$$P45= 000000        S2.ORQ= 000100
K$$LDC= 000000        N$LV1   000002        PR$LCC= 000010        P$$WRD= 000000        S2.PCU= 010000
K$$TPS= 000074        N$LV2   000010        PR$MOP= 000004        Q$$OPT= 000010        S2.RCU= 000001
LD$LP = 000000        N$MBXQ  000050        PR$UP = 000001        RCP1NM  000000R   002 S2.SRQ= 000040
LONACP  000010R   002 N$MHC1  000036        PR0   = 000000        RCP2NM  000004R   002 S2.WAL= 000010
LVL1  = 002000        N$MHC2  000044        PR1   = 000040        RF.CTL= 000003        S2.WRA= 000006
L$$ASG= 000000        N$PLD   000016        PR2   = 000100        RF.LD1= 000040        S2.WRB= 000002
L$$DRV= 000000        N$PLLT  000030        PR3   = 000140        RF.LD2= 000100        S3.ACR= 000001
L$$P11= 000001        N$PRI   000076        PR4   = 000200        RF.TIM= 177400        S3.CTC= 000004
L$$11R= 000000        N$ROA1  000022        PR5   = 000240        RF.TMO= 000400        S3.FDX= 000200
MPAR  = 172100        N$ROA2  000030        PR6   = 000300        RF.WFC= 000200        S3.ICE= 001000
MPCSR = 177746        N$RTMX  000014        PR7   = 000340        RF.WTD= 000020        S3.MHE= 000400
M$$CRB= 000124        N$RTM1  000014        PS    = 177776        RF.WTM= 000030        S3.NEC= 000020
M$$CRX= 000000        N$RTM2  000015        PS$CHR= 000016        RF.WTS= 000010        S3.PPT= 020000
M$$FCS= 000000        N$RT1   000000        PS$FAI= 000014        R$$DER= 000000        S3.PTH= 004000
M$$MGE= 000000        N$RT2   000006        PS$NTI= 000006        R$$K11= 000001        S3.RAL= 000010
M$$NET= 000000        N$SLA   000016        PS$OFF= 000000        R$$SND= 000000        S3.RES= 010000
M$$OVR= 000000        N$SNOD  000012        PS$STR= 000002        R$$11M= 000000        S3.RUB= 040000
NF$ACC= 100000        N$TCTL  000112        PS$UP = 000012        R$$11S= 000001        S3.TAB= 000002
NF$BLK= 000100        N$TIM   000004        PS$VER= 000010        R.UNGC= 000015        S3.TME= 002000
NF$DMO= 000010        N$TLC   000100        PS$WT = 000004        R.UNPC= 000014        S3.TSY= 000040
NF$EVT= 040000        N$TNC   000114        PT$BRO= 000200        R.UNPN= 000006        S3.8BC= 000100
NF$MOU= 000040        N$TRC   000106        PT$DRT= 000100        R.UNPR= 000012        S4.ABD= 100000
NF$RST= 000002        N$TTCB  000110        PT$END= 000004        R.UNRM= 000022        S4.ANI= 000400
NF$SCN= 000020        N$VCB   000010        PT$LV1= 000002        R.UNRU= 000024        S4.AVO= 001000
NF$SHU= 000004        N$VER   000066        PT$LV2= 000001        R.UNSM= 000016        S4.BLK= 002000
NF$TIM= 000200        N$XLEN  000124        PT$PH3= 000010        R.UNSU= 000020        S4.DEC= 004000
```

```
                                                                 G  6
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:31  Page 10-3
```

```
000424  000461                           BR      60$                     ;   AND RETURN TO COMMON CODE

000426  032765  002000  000000  50$:     BIT     #LF.ENA,L.FLG(R5)       ;SHOULD THIS LINE BE ENABLED?
000434  001455                           BEQ     60$                     ;NO
000436  011246                           MOV     (R2),-(SP)              ;GET REVERSE MAPPING VALUE
000440  000316                           SWAB    (SP)                    ;CALCULATE OWNER LLC INDEX
000442  005001                           CLR     R1                      ; ...
000444  111601                           MOVB    (SP),R1                 ; ...
000446  005726                           TST     (SP)+                   ;CLEAN UP STACK
000450  067701  000000G                  ADD     @PDVTA,R1               ;GET LLC PDV ADDRESS
000454  011101                           MOV     (R1),R1                 ; ...
000456  022761  114224  000004           CMP     #^RXPT,Z.NAM(R1)        ;XPT ?
000464  001030                           BNE     56$                     ;IF NO - BRANCH

000466  016701  000000G                  MOV     $DBECL,R1               ;COPY ECL DATA BASE ADDRESS
000472  132761  000040  000005           BITB    #NF$MOU,N$FLG(R1)       ;IS NETWORK ON ?
000500  001001                           BNE     51$                     ;IF YES - BRANCH
000502  000432                           BR      60$                     ;IF NO - DO NOT ENABLE CIRCUIT

000504  017701  000000G         51$:     MOV     @DECPT,R1               ;GET DECNET HOME BLOCK ADDRESS
000510  005761  000060                   TST     D$NN(R1)                ;ROUTING NODE ?
000514  001014                           BNE     56$                     ;IF YES - BRANCH
000516  005767  177262                   TST     ACTCIR                  ;OTHER LINES ENABLED ?
000522  001407                           BEQ     55$                     ;IF NO - BRANCH
000524  042765  002000  000000           BIC     #LF.ENA,L.FLG(R5)       ;DISABLE LINE
000532                                   EMSG$   12                      ;PRINT MSG
000540  000413                           BR      60$                     ;CONTINUE

000542  005267  177236          55$:     INC     ACTCIR                  ;COUNT ANOTHER LINE ENABLED
000546                          56$:     CALL    ENALIN                  ;ENABLE THE LINE
000552  103406                           BCS     60$                     ;BR IF SPECIFIED LLC DOES NOT OWN LINE
000554  052765  100000  000000           BIS     #LF.ACT,L.FLG(R5)       ;SET LINE ACTIVE
000562  112765  000000  000020           MOVB    #LN.ON,L.NMST(R5)       ;SET MANAGEMENT STATE TO 'ON'

000570  062702  000002          60$:     ADD     #2,R2                   ;NEXT SYSTEM LINE NUMBER
000574  005316                           DEC     (SP)                    ;REDUCE SYSTEM LINE COUNT
000576  001402                           BEQ     65$                     ;ANY MORE TO SCAN?
000600                                   CALLR   ENAC                    ;IF YES - CHECK NEXT CIRCUIT

000604  005726                  65$:     TST     (SP)+                   ;NO ... CLEAN UP THE STACK

000606                          70$:     RETURN
```

Table of contents

```
     6-   54    MACRO DEFINITIONS
     7-   84    LOCAL DATA
     8-  159    ERROR MESSAGES
     9-  259    NETWORK INITIALIZATION COLD START
    10-  334    MAIN CONTROL
    11-  469    VECSET - SET UP PROCESS VECTOR TABLE
    12-  539    NTL COMMUNICATION
    13-  568    FIND PDV INDICES OF NSP PROCESSES (ECL AND XPT)
    14-  613    ERROR MESSAGE PROCESSOR
    16-  747    RCV DATA AST
    17-  772    RCV DATA SUBROUTINE
    18-  831    PUTAST - PUT COMPLETION AST IN CALLER TASK'S AST QUEUE
```

```
                MOV     (SP)+,R0        ; COPY UNMAPPED ADDRESS FOR $DEACX

                MOV     (R5),-(SP)      ; SAVE NEXT BLOCK ADDRESS
                MOV     #V.LEN,R1       ; GET VECTOR BLOCK LENGTH
                CALLX   $DEACX,AUX      ; DEALLOCATE BLOCK
                MOV     (SP)+,R5        ; GET NEXT BLOCK
                BNE     10$

        15$:    RESMAP                  ; RESTORE MAPPING
        20$:    RETURN

                .ENDC

```

```
NTINIT11S  CREATED BY  MACRO  ON 5-SEP-85 AT 11:10       PAGE 1

SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE              REFERENCES

ABTIM      000456 RG       #8-224
ALOCB      000436 RG       #8-216
BUFUMP   = 172354          #6-77
CALLX      000472 RG       #8-235
CCBGT      000502 RG       #8-239
CEACC      000510 RG       #8-242
CF.DDM   = 000002          #6-74
CF.DYN   = 000004          #6-74
CF.EIS   = 000010          #6-74
CF.FRK   = 100000          #6-74
CF.LOG   = 000020          #6-74
CF.MDM   = 000001          #6-74
CF.TIM   = 000400          #6-74
CLKHD      000452 RG       #8-222
CMODE    = 140000          #6-77
DEACB      000440 RG       #8-217
DECPT      000516 RG       #8-245
DEVHD      000460 RG       #8-225
EXEVEC     000424 R        #8-202        8-252
FE.CEX   = ******   GX      10-357
FMASK      000426 RG       #8-203        10-357
FNDPDV     000204 RG       #13-585
HEADR      000444 RG       #8-219
IO.WVB   = ******   GX      14-689
I$$AS    = ******           6-72         6-72          6-75          6-77
KISAR0   = 172340          #6-77
KISAR6   = 172354          #6-77         8-204
KSAR6      000430 RG       #8-204
LDPWF      000462 RG       #8-226
LF.ACT   = 100000          #6-76
LF.BRO   = 000400          #6-76
LF.BWT   = 000007          #6-76
LF.ENA   = 002000          #6-76
LF.LPB   = 001000          #6-76
LF.MDC   = 000100          #6-76
LF.MFL   = 004000          #6-76
LF.MTP   = 000020          #6-76
LF.PAC   = 000200          #6-76
LF.RDY   = 040000          #6-76
LF.REA   = 010000          #6-76
LF.SER   = 000040          #6-76
LF.TIM   = 000010          #6-76
LF.UNL   = 020000          #6-76
LF.X2P   = 000000          #6-76
LGSTT      000514 RG       #8-244        10-399
LLCAL      000524 RG       #8-248
LLCRS      000504 RG       #8-240
LLCTA      000526 RG       #8-249
LN.CLO   = 000000          #6-76
LN.DUM   = 000005          #6-76
LN.LOA   = 000004          #6-76
LN.LOO   = 000003          #6-76
```

```
NTINIT11S  CREATED BY  MACRO  ON 5-SEP-85 AT 11:10       PAGE 2
```

```
                                         .SBTTL  CLINS - INSERT ENTRY ON CLOCK QUEUE
                                ;+
                                ; *** - CLINS - INSERT ENTRY ON CLOCK QUEUE WITHOUT CHANGING APR5 MAPPING VALUE
                                ;
                                ; THIS ROUTINE SHOULD BE CALLED AT SYSTEM STATE - TO INHIBIT TASK SWITCHING
                                ;
                                ; INPUTS:
                                ;        R0 = ADDRESS OF THE CLOCK QUEUE ENTRY BLOCK
                                ;        R1 = HIGH ORDER HALF OF DELTA TIME
                                ;        R2 = LOW ORDER HALF OF DELTA TIME
                                ;        R4 = REQUEST TYPE
                                ;
                                ; OUTPUTS:
                                ;        THE CLOCK QUEUE ENTRY IS INSERTED IN THE CLOCK QUEUE ACCORDING
                                ;        TO THE TIME THAT IT WILL COME DUE.
                                ;
                                ;-
000162                          CLINS:
000162  005020                           CLR     (R0)+                   ;* POINT TO TYPE FIELD AND CLEAR LINK
000164  110420                           MOVB    R4,(R0)+                ;* SET REQUEST TYPE
000166  005200                           INC     R0                      ;* ADVANCE TO TCB ADDRESS
000170  005720                           TST     (R0)+                   ;* SKIP TCB LOCATION
000172  067702  000000G                  ADD     @ABTIM,R2               ;* CALCULATE ABSOLUTE TIME WHEN
000176  005501                           ADC     R1                      ;* REQUEST WILL COME DUE
000200  010220                           MOV     R2,(R0)+                ;* SET LOW ORDER PART OF TIME
000202  010110                           MOV     R1,(R0)                 ;* SET HIGH ORDER PART OF TIME
000204  016702  000000G                  MOV     CLKHD,R2                ;* POINT TO CLOCK QUEUE LIST HEAD
000210  010201                  10$:     MOV     R2,R1                   ;* SAVE ADDRESS OF PREVIOUS ENTRY
000212  011102                           MOV     (R1),R2                 ;* GET ADDRESS OF NEXT ENTRY
000214  001410                           BEQ     20$                     ;* IF EQ, END OF LIST
000216  021062  000010                   CMP     (R0),C.TIM+2(R2)        ;* ENTRY GO HERE ?
000222  101372                           BHI     10$                     ;* IF HI, NO
000224  103404                           BLO     20$                     ;* IF LO, ENTRY GOES HERE
000226  026062  177776  000006           CMP     -2(R0),C.TIM(R2)        ;* REALLY GO HERE ?
000234  103365                           BHIS    10$                     ;* IF HIS, NO
000236  162700  000010          20$:     SUB     #8.,R0                  ;* BACK UP TO FRONT OF BLOCK
000242  010011                           MOV     R0,(R1)                 ;* SET NEW NEXT IN PREVIOUS ENTRY
000244  010210                           MOV     R2,(R0)                 ;* SET NEXT IN NEW ENTRY
000246                                   RETURN
```

```
NTIUMR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:33      PAGE 2
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

RETURN            7-100
SCBDF$           #6-59         6-61
```

Table of contents

```
     5-  122     $CFDEF - PROCESS DEFINE COMMAND
     6-  139     SETEXE - SET EXECUTOR PARAMETERS
     7-  233     SETLIN - SET LINE PARAMETERS
     8-  261     SETCIR - SET CIRCUIT PARAMETERS
     8-  262     SETKNC - SET KNOWN CIRCUIT PARAMETERS
     9-  333     SETTRN - SET TRANSPORT CIRCUIT PARAMETERS
    10-  361     SETSVC - SET SVC PARAMETERS
    10-  362     SETPVC - SET PVC PARAMETERS
    11-  399     SETSYS - PROCESS ''SET SYSTEM'' REQUEST
    12-  421     SETXPR - SET MODULE X25-PROTOCOL PARAMETERS
```

```
                                        .SBTTL  SETXPR - SET MODULE X25-PROTOCOL PARAMETERS
                                ;+
                                ; ** SETXPR - SET MODULE X25-PROTOCOL PARAMETERS
                                ;
                                ; INPUT         $OPRND = PARSED PARAMETERS
                                ;
                                ; OUTPUT        THE PROPER ROUTINE IS DISPATCHED TO
                                ;-

001226                          SETXPR:
001226  016700  000000G                 MOV     $PSIPT,R0       ; POINT TO PSN$DF
001232  001435                          BEQ     101$            ; IF EQ, TEMPLATE MISSING
001234  032767  000000G 000000G         BIT     #XP.NNA,$OPRND  ; CHANGE NETWORD NAME ?
001242  001406                          BEQ     5$              ; IF EQ, NO
001244  016760  000000G 000000G         MOV     $NNAM,P$SNAM(R0); UPDATE NETWORK NAME
001252  016760  000002G 000002G         MOV     $NNAM+2,P$SNAM+2(R0) ; ...
001260  032767  000000C 000000G 5$:     BIT     #<XP.DTE!XP.KND>,$OPRND ; DTE PARAMETERS ?
001266  001402                          BEQ     10$             ; IF EQ, NO
001270  004767  000000G                 CALL    $SEDTE          ; DO DTE OPERATIONS
001274  032767  000000C 000000G 10$:    BIT     #<XP.CUG!XP.KNG>,$OPRND ; GROUP PARAMETERS ?
001302  001402                          BEQ     20$             ; IF EQ, NO
001304  004767  000000G                 CALL    $SECUG          ; DO GROUP OPERATION
001310  032767  000000G 000000G 20$:    BIT     #XP.LV3,$OPRND  ; UPDATE LEVEL 3 CHARACTERISTICS?
001316  001402                          BEQ     30$             ; IF EQ, NO
001320  004767  000000G                 CALL    $SEX3P          ; UPDATE LEVEL 3
001324  000207                  30$:    RETURN

001326                          101$:   ERROR$  #ERRPSI         ; YES WE HAVE NO PSI
001336  000261                          SEC
001340  000207                          RETURN

        000001                          .END
```

Table of contents

```
     5-  103    $CFHLP - PROCESS HELP COMMAND
```

```
    123                                              .SBTTL  GLOBAL DATA
    124                                      ;
    125                                      ; DPB'S
    126                                      ;
    127 000354                               $TIDPB::QIOW$   IO.WVB,LUN2,LUN2,,,,<0,0,40>
    128 000404                               $ERDPB::QIOW$   IO.WVB,LUN3,LUN3,,,,<0,0,40>
    129
    130 000434                               $BUFF:: .BLKB   BUFSIZ          ; TERMINAL BUFFER (TWICE THE NORMAL BUFFER SIZE)
    131
    132 001420  000000  001420'              $TMLST::.WORD   0,.-2           ; LISTHEAD OF TEMPLATES IN END OF TASK BUFFER
    133 001424                               $PSIPT::.BLKW   1               ; ADDRESS OF PSN$DF TEMPLATE
    134 001426                               $SNAPT::.BLKW   1               ; ADDRESS OF SNA$DF TEMPLATE
    135 001430  000000                       $SVUPD::.WORD   0               ; SVC UPDATE FLAGS
    136 001432  000000                       $FILHD::.WORD   0               ; file descriptor listhead
    137 001434  000000                       $LSTMP::.WORD   0               ; POINTER TO LAST TEMPLATE IN LIST (FOR REM$DF)
    138 001436  000000                       $LSNAD::.WORD   0               ; NODE ADDRESS OF LAST REM$DF
    139
    140 001440  000000                       $CNFIG::.WORD   0               ; CONFIGURATION FLAGS
    141 001442                               $FLAGS::.BLKW   2               ; SPECIAL FLAGS
    142 001446                               $CMAND::.BLKW   1               ; COMMAND CODE
    143 001450                               $COMP:: .BLKW   1               ; COMPONENT TYPE
    144 001452                               $OPRND::.BLKW   1               ; OPERANDS PRESENT
    145 001454                               $OPRN2::.BLKW   2               ; MORE OPERANDS PRESENT
    146 001460                               $SWICH::.BLKW   1               ; SWITCHES (ON/OFF)
    147 001462                               $WILD:: .BLKW   1               ; WILD CARDS IN EFFECT
    148 001464                               $INFO:: .BLKB   1               ; Information type for LIST:
    149         000000                               IN$SUM  == 0            ;       summary
    150         000002                               IN$STA  == 2            ;       status
    151         000004                               IN$CHA  == 4            ;       characteristics
    152         000006                               IN$COU  == 6            ;       counters
    153         000010                               IN$EVE  == 10           ;       events
    154 001465                                       .BLKB   1               ; (available)
    155 001466                               $CIRCT::.BLKW   1               ; CIRCUIT ID PARSE FLAG
    156 001470                               $EXSTS::.BLKW   1               ; EXIT STATUS
    157 001472                               $HIVBN::.BLKW   1               ; SAVE THE VBN AND BYTE ADDRESS OF
    158 001474                               $LOVBN::.BLKW   1               ;   THE INPUT FILE TO POSITION AT
    159 001476                               $BYTE:: .BLKW   1               ;   TOP OF FILE.
    160 001500                               $NTNAM::.BLKW   2               ; EXECUTOR NODE NAME (RAD50)
    161 001504                               $NTADD::.BLKW   1               ; EXECUTOR ADDRESS
    162
    163 001506  000000                       $BYTMN::.WORD   0               ; POOL BYTE EXTENSION MINIMUM VALUE
    164 001510  000000                       $BYTXT::.WORD   0               ; POOL BYTE EXTENSION CURRENT VALUE
    165
    166 001512                               .RNFSP::                        ; ADDRESS OF FILESPEC (REMOTE NODE PARSING)
    167 001512  000000                       $UMRIN::.WORD   0               ; NON-UMR MAPPED BYTES, INITIAL
    168 001514  000000                       $UMRUP::.WORD   0               ; NON-UMR MAPPED BYTES, UPDATED
    169 001516  177777                       $PBLK:: .WORD   -1              ; FUDGE FACTOR IN POOL BLOCK ALLOCATION
    170                                      ;
    171                                      ; COMPONENT OPTIONS
    172                                      ;
    173 001520                               $CCBNM::.BLKW   1               ; NUMBER OF CCB'S
    174 001522                               $RDBNM::.BLKW   1               ; NUMBER OF RDB'S
    175 001524                               $RDBSZ::.BLKW   1               ; SIZE OF EACH RDB
    176 001526                               $SDBNM::.BLKW   1               ; THE NUMBER OF SDB'S
    177 001530                               $THRSH::.BLKW   1               ; RDB THRESHOLD VALUE
    178                                      ;
    179                                      ; LINE DESCRIPTOR - MUST BE CONTIGUOUS
```

```
NTIUMR  MACRO V05.03b Saturday 29-Jun-85 03:28  Page 7
LOAD UMR'S

    102         000001                          .END
```

```
NTIUMR  MACRO V05.03b Saturday 29-Jun-85 03:28  Page 7-1
Symbol table
```

```
                                                                  G  2
NTIX25     CREATED BY  MACRO  ON 29-JUN-85 AT 03:29      PAGE 1

SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

ALOCB    = ******  GX        7-148
ASSIGN     000026 R         #6-89       7-125
A.LULU   = 000002            6-89
BUFUMP   = 172354           #5-59
CMODE    = 140000           #5-59
DEACB    = ******  GX        7-185
HEADR    = ******  GX        7-129
H.LUN      000076            7-130
I$$AS    = ******            5-55       5-59        5-60
KISAR0   = 172340           #5-59
KISAR6   = 172354           #5-59
LONACP     000000 R         #6-88       7-156
MPAR     = 172100           #5-59
MPCSR    = 177746           #5-59
M$$MGE   = 000000            5-59
N$$VCT   = ******            7-165
PDVID    = ******  GX        7-163
PIRQ     = 177772           #5-59
PMODE    = 030000           #5-59
PR0      = 000000           #5-59
PR1      = 000040           #5-59
PR2      = 000100           #5-59
PR3      = 000140           #5-59
PR4      = 000200           #5-59
PR5      = 000240           #5-59
PR6      = 000300           #5-59
PR7      = 000340           #5-59
PS       = 177776           #5-59
R$R0     = 000002           #6-76
R$R1     = 000004           #6-77
R$R2     = 000006           #6-78
R$R3     = 000010           #6-79
R$R4     = 000012           #6-80
R$R5     = 000014           #6-81
R$$MPL   = ******            5-59       5-59        5-59        7-128
R$$11D   = ******            5-59       5-60
R$$11M   = 000000            5-60
R$$11S   = ******            5-60       7-159
R.UNTN   = 000002            6-88
SWR      = 177570           #5-59
S$$BAS   = ******            6-95       6-95        6-96        6-96        6-97        6-97
TPS      = 177564           #5-59
UBMPR    = 170200           #5-59
UISAR0   = 177640           #5-59
UISAR1   = 177642           #5-59
U.VCB    = 000034            7-142     *7-154      *7-180
V$LEN      000034            7-144      7-182
X$$MCB   = ******            5-60       5-60
ZF.COU   = 001000           #5-60
ZF.DDM   = 000001           #5-60
ZF.DIA   = 004000           #5-60
ZF.DLC   = 000002           #5-60
```

```
                                                                  H  2
NTIX25     CREATED BY  MACRO  ON 29-JUN-85 AT 03:29      PAGE 2
```

```
                                                                   G  3
NTICEX  MACRO V05.03b Saturday 29-Jun-85 03:29  Page 9-1
Symbol table

A$$CHK= 000000      CS.DIS= 000040      D$END = 000104      FS.HLT= 000000          LF.UNL= 020000
A$$CPS= 000000      CS.ENA= 000001      D$FNB   000034      FS.INI= 000000          LF.X2P= 000000
A$$PRI= 000000      CS.ENB= 000020      D$HIOR  000024      FS.KIL= 000000          LLCAL = ******  GX
A$$TRP= 000000      CS.ERR= 100000      D$HOST  000022      FS.LCL= 100000          LN.CLO= 000000
BUFUMP= 172354      CS.FTL= 001000      D$INAC  000044      FS.LTM= 001000          LN.DUM= 000005
CB.CCB= 000002      CS.HCR= 000001      D$INCT  000042      FS.MNT= 004000          LN.LOA= 000004
CB.DDM= 000040      CS.HFE= 002000      D$IPL   000051      FS.MSN= 014000          LN.LOO= 000003
CB.DLC= 000020      CS.LST= 040000      D$LID   000020      FS.REA= 001000          LN.OAU= 000003
CB.RDB= 000004      CS.MTL= 004000      D$LNAM  000006      FS.RET= 000000          LN.OFF= 000001
CB.SDB= 000010      CS.RNG= 000010      D$LNUM  000014      FS.REZ= 003000          LN.ON = 000000
CB.SLI= 000100      CS.ROV= 000004      D$LST   000047      FS.RLB= 002000          LN.OOP= 000004
CB.XLB= 000001      CS.RSN= 010000      D$MAXC  000064      FS.RNG= 011000          LN.OPE= 000001
CC.LLC= 000200      CS.SHU= 000001      D$MAXH  000066      FS.RST= 000000          LN.REF= 000002
CE.ABO= 100362      CS.SID= 000002      D$MAXV  000070      FS.RTN= 001000          LN.SER= 000002
CE.DAO= 100346      CS.STR= 000004      D$MLL   000040      FS.SET= 005000          LN.STA= 000017
CE.DIS= 100366      CS.SUC= 000001      D$MNOD  000041      FS.SFC= 005000          LN.SUB= 000360
CE.ERR= 100370      CS.TMO= 020000      D$NA    000062      FS.SFR= 006000          LN.TRI= 000006
CE.ILN= 100350      CS.XUR= 000004      D$NBEA  000056      FS.SFS= 004000          L$$ASG= 000000
CE.LTO= 100356      C$$CKP= 000000      D$NBRA  000054      FS.SPW= 040000          L$$DRV= 000000
CE.MOP= 100372      C$$ORE= 000400      D$NEND= 000054      FS.STM= 000000          L$$P11= 000001
CE.NTE= 100361      C$$RSH= 177564      D$NLN   000030      FS.STP= 002000          L$$11R= 000000
CE.RTE= 100376      C.ADD   000034      D$NN    000060      FS.STR= 001000          L.COST  000015
CE.SRC= 100364      C.BID   000003      D$OUTT  000043      FS.TRM= 003000          L.CTL   000012
CE.STP= 100352      C.BUF   000014      D$RETF  000050      FS.WLB= 001000          L.CVA   177776
CE.TME= 100354      C.BUF1  000014      D$RNN   000002      FS.XKL= 002000          L.DDM   000002
CE.TMO= 100374      C.BUF2  000024      D$RTMR  000076      FS.XOF= 010000          L.DDS   000004
CE.UNS= 100344      C.CNT   000020      D$SEG   000036      FS.XON= 007000          L.DLC   000003
CF.CHN= 000001      C.CNT1  000020      D$SER   000032      FS.ZER= 002000          L.DLM   000006
CF.EOM= 000004      C.CNT2  000030      D$SQRL  000052      F$$LVL= 000001          L.DLS   000010
CF.HDR= 000020      C.FLG   000022      D$$BUG= 177514      G$$TPP= 000000          L.FLG   000000
CF.LB = 100000      C.FLG1  000022      D$$ISK= 000000      G$$TSS= 000000          L.KRBA  000016
CF.LIN= 000002      C.FLG2  000032      D$$L11= 000001      G$$TTK= 000000          L.LEN = 000022
CF.SOM= 000010      C.FNC   000010      D$$YNC= 000000      G$$WRD= 000000          L.MPF   000022
CF.SYN= 000040      C.LIN   000006      D$$YNM= 000000      INIFUN  000000R     002 L.NMST  000020
CF.TRN= 000100      C.LNK   000000      E$$XPR= 000000      I$$RAR= 000000          L.NSTA  000014
CMODE = 140000      C.MOD   000011      FC.CCP= 000020      I$$RDN= 000000          L.OWNR  000021
CM.CIR= 000002      C.NSP   000004      FC.CTL= 000006      KISAR0= 172340          L.UNT   000013
CM.FMT= 100000      C.PRO   000042      FC.KCP= 000016      KISAR6= 172354          MPAR  = 172100
CM.HRD= 000002      C.RSV   000002      FC.KIL= 000004      KSAR6 = ******  GX      MPCSR = 177746
CM.LIN= 000000      C.STA   000007      FC.MAN= 000024      K$$CNT= 1775[illegible]          M$$CRB= 000124
CM.LOO= 000001      C.STS   000012      FC.MLD= 000026      K$$CSR= 177[illegible]           M$$CRX= 000000
CM.XLO= 000004      C.URM   177776      FC.PCT= 000030      K$$LDC= 000000          M$$EXT= 000000  G
CP.DCF= 000040      C.XACP  000004      FC.PWR= 000022      K$$TPS= 000074          M$$FCS= 000000
CP.HDL= 000007      C.XID   000035      FC.RCE= 000002      LD$LP = 000000          M$$MGE= 000000
CP.PS = 177400      C.XLEN  000044      FC.RCP= 000014      LF.ACT= 100000          M$$NET= 000000
CP.PSI= 000200      C.XPL   000040      FC.TIM= 000010      LF.BRO= 000400          M$$OVR= 000000
CP.XCF= 000100      C.XPT   000034      FC.XCP= 000012      LF.BWT= 000007          NTILIN  000000RG
CP.2FR= 000030      C.XS[illegible]C  000042      FC.XME= 000000      LF.ENA= 002000          NTLPT = ******  GX
CS.ABO= 000100      C.XTC   000037      FNDPDV= ******  GX  LF.LPB= 001000          N$$ACC= 000001
CS.BRO= 000002      C.X25   000036      FS.AST= 000000      LF.MDC= 000100          N$$BUF= 000001
CS.BUF= 000200      D$AMXC  000072      FS.CIB= 002000      LF.MFL= 004000          N$$LDV= 000001
CS.CES= 000002      D$AMXH  000074      FS.CRA= 001000      LF.MTP= 000020          N$$MCP= 000001
CS.CHN= 000010      D$ANN   000000      FS.DIS= 013000      LF.PAC= 000200          N$$MLL= 000001
CS.CMP= 000200      D$BRPR  000102      FS.DVC= 001000      LF.RDY= 040000          N$$MOV= 000010
CS.DCR= 000400      D$BRTM  000100      FS.ENB= 012000      LF.REA= 010000          N$$NCT= 000001
CS.DEF= 000004      D$DELF  000045      FS.EXI= 001000      LF.SER= 000040          N$$PEM= 000001
CS.DEV= 000002      D$DELW  000046      FS.GET= 006000      LF.TIM= 000010          PDVID = ******  GX
```

```
                                                                   H  3
NTICEX  MACRO V05.03b Saturday 29-Jun-85 03:29  Page 9-2
```

```
                                                                                  G  4
NTICSR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00
SYMBOL   VALUE               REFERENCES

BUFJMP  = 172354             #6-61
CMODE   = 140000             #6-61
CSRDIS    000006 R           #6-75         7-151
CSRENA    000000 R           #6-74         7-149
CSRSLT    000016 R           #6-82        *7-134        7-157
CSRTAB    000014 R            6-74        #6-77
CSRTRP    000250 R            6-77        #7-204
I$$AS   = ******              6-61         6-63
KISAR0  = 172340             #6-61
KISAR6  = 172354             #6-61
KSAR6   = ******    GX        7-124        7-129
LF.ACT  = 100000             #6-62
LF.BRO  = 000400             #6-62
LF.BWT  = 000007             #6-62
LF.ENA  = 002000             #6-62         7-171
LF.LPB  = 001000             #6-62
LF.MDC  = 000100             #6-62
LF.MFL  = 004000             #6-62
LF.MTP  = 000020             #6-62
LF.PAC  = 000200             #6-62
LF.RDY  = 040000             #6-62         7-171
LF.REA  = 010000             #6-62
LF.SER  = 000040             #6-62
LF.TIM  = 000010             #6-62
LF.UNL  = 020000             #6-62
LF.X2P  = 000000             #6-62
LN.CLO  = 000000             #6-62
LN.DUM  = 000005             #6-62
LN.LOA  = 000004             #6-62
LN.LOO  = 000003             #6-62
LN.OAU  = 000003             #6-62
LN.OFF  = 000001             #6-62
LN.ON   = 000000             #6-62
LN.OOP  = 000004             #6-62
LN.OPE  = 000001             #6-62
LN.REF  = 000002             #6-62
LN.SER  = 000002             #6-62
LN.STA  = 000017             #6-62
LN.SUB  = 000360             #6-62
LN.TRI  = 000006             #6-62
L.COST    000015             #6-62
L.CTL     000012             #6-62         7-186
L.CVA     177776             #6-62
L.DDM     000002             #6-62         7-162        7-187
L.DDS     000004             #6-62
L.DLC     000003             #6-62
L.DLM     000006             #6-62
L.DLS     000010             #6-62
L.FLG     000000             #6-62        *7-171
L.KRBA    000016             #6-62
L.LEN   = 000022             #6-62
L.MPF     000022             #6-62
```

```
                                                                                  H  4
NTICSR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 2
```

```
**FILE**ID**NTILON




NN      NN  TTTTTTTTTT    IIIIII    LL            000000    NN      NN
NN      NN  TTTTTTTTTT    IIIIII    LL            000000    NN      NN
NN      NN      TT          II      LL          00      00  NN      NN
NN      NN      TT          II      LL          00      00  NN      NN
NNNN    NN      TT          II      LL          00      00  NNNN    NN
NNNN    NN      TT          II      LL          00      00  NNNN    NN
NN  NN  NN      TT          II      LL          00      00  NN  NN  NN
NN  NN  NN      TT          II      LL          00      00  NN  NN  NN
NN    NNNN      TT          II      LL          00      00  NN    NNNN
NN    NNNN      TT          II      LL          00      00  NN    NNNN
NN      NN      TT          II      LL          00      00  NN      NN  ....
NN      NN      TT          II      LL          00      00  NN      NN  ....
NN      NN      TT        IIIIII    LLLLLLLLLL    000000    NN      NN  ....
NN      NN      TT        IIIIII    LLLLLLLLLL    000000    NN      NN  ....




    11          11        SSSSSSSS
    11          11        SSSSSSSS
  1111        1111      SS
  1111        1111      SS
    11          11      SS
    11          11      SS
    11          11        SSSSSS
    11          11        SSSSSS
    11          11              SS
    11          11              SS
    11          11              SS
    11          11              SS
  111111      111111    SSSSSSSS
  111111      111111    SSSSSSSS
```

```
Symbol table

S4.DLO= 000100          T$T5    000030          U.ACB   000062          U.TUX   000024          W.BHVR  000004
S4.EDT= 010000          T$T6    000032          U.ACP = 000032          U.UHVR= 000107          W.BLGH  000020
S4.HFF= 000020          T$$KMG= 000000          U.ADMA  000066          U.UIC   000044          W.BLPD  000016
S4.HFL= 000007          T$$MIN= 000000          U.AFLG  000064          U.UNFL= 000052          W.BLVR  000002
S4.HHT= 000040          UBMPR = 170200          U.ATT   000022          U.UNIT  000006          W.BNPD  000015
S4.HSY= 000200          UC.ALG= 000200          U.BPKT= 000044          U.UNSZ= 000114          W.BOFF  000012
S4.RGS= 020000          UC.ATT= 000010          U.BUF   000024          U.UNTI= 000060          W.BPCB  000000
S4.SFC= 040000          UC.KIL= 000004          U.CBF = 000032          U.USVR= 000106          W.BSIZ  000010
S4.VFL= 000010          UC.LGH= 000003          U.CLI   177772          U.UTIL= 000036          X$$DBT= 000000
S5.ABP= 100000          UC.NPR= 000100          U.CNT   000030          U.VCB = 000034          ZF.COU= 001000
S5.BCC= 020000          UC.PWF= 000020          U.COTQ  000030          U.VSER= 000120          ZF.DDM= 000001
S5.DAO= 040000          UC.QUE= 000040          U.CTCB  000026          U.2MED= 000070          ZF.DIA= 004000
S5.HPC= 000014          UD.UNS= 000000          U.CTL   000004          U2.AT.= 000020          ZF.DLC= 000002
S5.HPO= 000020          UD.160= 000004          U.CTYP  000052          U2.CRT= 002000          ZF.DVP= 100000
S5.ITI= 000100          UD.200= 000001          U.CW1   000010          U2.DH1= 100000          ZF.INI= 040000
S5.OXF= 000040          UD.556= 000002          U.CW2   000012          U2.DJ1= 040000          ZF.KMX= 000020
S5.RPO= 002000          UD.625= 000005          U.CW3   000014          U2.DZ1= 000100          ZF.LLC= 000004
S5.SW1= 000001          UD.8K = 000006          U.CW4   000016          U2.ESC= 001000          ZF.LMC= 000100
S5.TMM= 000002          UD.800= 000003          U.CYL = 000104          U2.HFF= 010000          ZF.MAN= 020000
S5.VER= 010000          UISAR0= 177640          U.DCB   000000          U2.HLD= 000040          ZF.MFL= 000010
S5.XOF= 000004          UISAR1= 177642          U.FCDE= 000042          U2.LOG= 000400          ZF.MTM= 000400
S5.XON= 000010          UM.CLI= 000036          U.FNUM= 000040          U2.LWC= 000001          ZF.MUX= 000040
S6.EIO= 000400          UM.DSB= 000200          U.GRP = 000102          U2.L3S= 000004          ZF.PSE= 002000
S6.RDI= 100000          UM.NBR= 000400          U.KCSR= 000032          U2.L8S= 010000          ZF.SLI= 010000
S6.RLU= 001000          UM.OVR= 000001          U.KCS6= 000034          U2.NEC= 004000          ZF.TIM= 000200
TPS   = 177564          US.ABO= 000001          U.LUIC  177774          U2.PRV= 000010          ZF.X3P= 000000
T$FLAG  000044          US.BSP= 000002          U.MEDI= 000040          U2.RMT= 020000          ZS.ASN= 100000
T$LIF   000013          US.BSY= 000200          U.MLUN= 000050          U2.R04= 100000          ZS.BSY= 140000
T$LIFL  000013          US.FOR= 000040          U.MUP   177772          U2.SCS= 000004          Z.AVL   000014
T$LIFO  000013          US.FRK= 000002          U.OWN   177776          U2.SLV= 000200          Z.DAT   000016
T$LIFS  000013          US.KPF= 000001          U.RBNS= 000112          U2.VT5= 000002          Z.DSP   000000
T$LIN   000000          US.LAB= 000004          U.RCTC= 000113          U2.7CH= 010000          Z.FLG   000010
T$LIPS  000006          US.MDE= 000002          U.RCTS= 000110          U3.DPA= 100000          Z.LEN = 000016
T$LLD   000012          US.MDM= 000020          U.RED   000002          U3.PAR= 040000          Z.LLN   000006
T$LLDC  000045          US.MNT= 000100          U.RED2  000034          U3.UPC= 020000          Z.MAP   000020
T$LLDL  000012          US.OFL= 000001          U.SCB   000020          U4.CR = 000100          Z.NAM   000004
T$LLDO  000012          US.PUB= 000004          U.SHST= 000076          VF$RCV= 100000          Z.PCB   000012
T$LLDS  000012          US.PWF= 000010          U.SHUN= 000074          VF$XMT= 040000          Z.SCH   000007
T$LLEN  000046          US.RED= 000002          U.SPC = 000036          V$FLG   000000          $DBECL= ****** GX
T$LOPR  000002          US.SHR= 000001          U.STS   000005          V$LEN   000022          $DBXPT= ****** GX
T$LTCL  000024          US.SPU= 000002          U.ST2   000007          V$RCV   000002          $ERR11  000046RG      002
T$LTIM  000026          US.UMD= 000010          U.SUB = 000036          V$XMT   000012          $ERR15  000136RG      002
T$LTPR  000014          US.VV = 000001          U.TCHP  000042          V$$CTR= 001000          $ERR17  000176RG      002
T$LTPS  000020          US.WCK= 000010          U.TCVP  000043          V.CULT  000004          $ERR18  000252R       002
T$NAPL  000004          UU.ABO= 000400          U.TFLK  000040          V.GENQ  000016          $ERR19  000312RG      002
T$NFE   000000          UU.ATN= 000100          U.TFRQ  000036          V.MBXQ  000012          $NIERR= ****** GX
T$NLEN  000010          UU.AVN= 000004          U.TIXL  000060          V.TIME  000003          $NILON  000000RG
T$NNUL  000002          UU.GUS= 000010          U.TLPP  000053          V.TIMR  000002          $NIMFE= ****** GX
T$NOPL  000006          UU.IOS= 002000          U.TMTI  000047          V.TRCT  000000          $NIXPT= ****** GX
T$NRNI  000042          UU.ONL= 000020          U.TRCK= 000100          V.ULT   000006          $$$OST= 000010
T$NRPL  000005          UU.RCT= 000002          U.TSTA  000026          V.ULTE  000010          $$$T1 = 000002
T$NRUL  000007          UU.RDY= 000200          U.TST5  000054          V.VBL = 000022          $$$T2 = 000042R       002
T$NVR   000001          UU.SER= 000001          U.TST6  000056          W.BATT  000006          .BASEB= 140000
T$RPRI  000040          UU.SIO= 001000          U.TTAB  000050          W.BFPD  000014          .$$$$.= 000034
T$SVC   000034          UU.SPC= 000040          U.TTYP  000046

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
```

```
                                                .SBTTL   ENABLE A LINE
                                        ;+
                                        ; **-ENALIN-ENABLE A LINE IF OWNED BY SPECIFIED LLC
                                        ;
                                        ; IF SO, ENABLE THE LINE BY QUEUEING A CCB TO THE PROPER LLC
                                        ;
                                        ; INPUTS:
                                        ;       R0 - PDV ADDRESS OF ASSOCIATED DDM
                                        ;       R2 - POINTER TO LINE'S ENTRY IN THE REVERSE MAPPING TABLE
                                        ;
                                        ; OUTPUTS:
                                        ;       START FUNCTION QUEUED TO OWNING LLC
                                        ;-

000610                                  ENALIN: SAVRG    <R0,R5>                 ;SAVE R3,R5
000614  010005                                  MOV      R0,R5                   ;GET PDV ADDRESS OF DDM
000616  005001                                  CLR      R1                      ;GET OWNING LLC PDV INDEX
000620  156201  000001                          BISB     1(R2),R1                ; ...
000624  111203                                  MOVB     (R2),R3                 ;GET CHANNEL #
000626  001447                                  BEQ      35$                     ;IF ZERO, BAD CHANNEL
000630  017700  000000G                         MOV      @PDVTA,R0               ;POINT TO PDV INDEX TABLE
000634  060100                                  ADD      R1,R0                   ;ADD OFFSET FOR SPECIFIED LLC
000636  011000                                  MOV      (R0),R0                 ;POINT TO LLC'S PDV

                                                .IF NDF  R$$11S
                                                BIT      #ZF.PSE,Z.FLG(R5)       ; IS THIS A DLM CIRCUIT?
                                                BEQ      10$                     ; BR IF NO
                                                CALL     DLMSET                  ; SET UP DATA LINK MAPPING PARAMETERS
                                                .ENDC

000640  016000  000016                  10$:    MOV      Z.DAT(R0),R0            ;GET POINTER TO LLC'S DATABASE
000644  001440                                  BEQ      35$                     ;IF ZERO, LLC NOT LOADED!!!

000646                                  15$:    SWSTK$   30$                     ;; ENTER SYSTEM STATE
000652                                          CALL     @CCBGT                  ;; TRY TO ALLOCATE A CCB
000656  103420                                  BCS      20$                     ;; IF CS, ALLOCATION FAILURE
000660  005064  000014                          CLR      C.BUF1(R4)              ;; CLEAR FLAGS WORD
000664  110164  000007                          MOVB     R1,C.STA(R4)            ;; SET UP PDV AND
000670  110364  000006                          MOVB     R3,C.LIN(R4)            ;;   CHANNEL # TO TURN ON
000674  005064  000004                          CLR      C.NSP(R4)               ;; CONTROL IS RETURNED TO AUX
000700  012764  000004  000012                  MOV      #STRT,C.STS(R4)         ;; TURN ON THE LINE
000706  012764  001006  000010                  MOV      #FC.CTL!FS.STR,C.FNC(R4);; SET UP FUNCTION CODE
000714                                          CALLR    @LLCRS                  ;; QUEUE CCB TO LLC
                                                                                                     ;**-1
000720                                  20$:    RETC     R0                      ;; SET USER C-BIT
000732                                          RETURN                           ;; LEAVE SYSTEM STATE

000734  103005                          30$:    BCC      40$                     ; IF CC, REQUEST QUEUED
000736                                          WSIG$S                           ; ALLOCATION FAILURE ... WAIT A BIT
000744  000740                                  BR       15$                     ;   THEN TRY AGAIN

000746  000261                          35$:    SEC                              ; INDICATE LINE NOT ENABLED
000750                                  40$:    RESRG    <R5,R0>                 ; RESTORE R5,R3
000754                                          RETURN
```

```
        .TITLE  NTINIT
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       NETWORK INITIALIZER (NTINIT) MAIN CONTROL
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
        .SBTTL  NTL COMMUNICATION

;+
; $NINTL - SEND DATA TO NTL AND WAIT FOR RESPONSE
;
; THIS ROUINE IS ONLY NECESSARY ON AN RSX11M OR IAS SYSTEM
;
; INPUTS:
;       $NTLFC IS ALREADY SET FOR NTL
;
; OUTPUTS:
;       CARRY INDICATES SUCCESS/FAILURE
;       $NTLFC CONTAINS RESPONSE FROM NTL
;-

        .IF NDF R$$11S                  ;*-RSX11M SYSTEM ONLY-*

$NINTL::
        DIR$    #$NISNL                 ;SEND DATA TO NTL
        BCS     10$                     ;BR IF CAN'T, PROBABLY NOT INSTALLED
        DIR$    #$NIRNL                 ;REQUEST NTL TO RUN IF NOT ALREADY
        WTSE$S  #15.                    ;WAIT FOR NTL'S RESPONSE
        CLEF$S  #15.                    ;CLEAR FLAG 15. FOR NEXT TIME
        RETURN                          ; WITH C-BIT CLEAR

10$:    EMSG$R  02                      ;NTL PROBABLY NOT INSTALLED

        .ENDC                           ;*-END RSX11M CODE-*
```

```
NTINIT11S  CREATED BY  MACRO  ON 5-SEP-85 AT 11:10       PAGE 2

SYMBOL  CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE             REFERENCES

LN.OAU  = 000003          #6-76
LN.OFF  = 000001          #6-76
LN.ON   = 000000          #6-76
LN.OOP  = 000004          #6-76
LN.OPE  = 000001          #6-76
LN.REF  = 000002          #6-76
LN.SER  = 000002          #6-76
LN.STA  = 000017          #6-76
LN.SUB  = 000360          #6-76
LN.TRI  = 000006          #6-76
LO.CON    000006   G      #6-72
LO.CSR    000006   G      #6-72
LO.CTL    000014   G      #6-72
LO.LIN    000004   G      #6-72
LO.PAR    000010   G      #6-72
LO.PRI    000010   G      #6-72
LO.SCR    000014   G      #6-72
LO.STA    000015   G      #6-72
LO.VCT    000012   G      #6-72
LR.AST    000026   G      #6-72
LR.CIR    000002   G      #6-72
LR.CTL    000004   G      #6-72
LR.EFN    000004   G      #6-72
LR.LIN    000002   G      #6-72
LR.PRM    000030   G      #6-72
LR.PRO    000002   G      #6-72
LR.STS    000000   G      #6-72
LR.TCB    000002   G      #6-72
LR.UNT    000005   G      #6-72
LS.AST  = 100000   G      #6-72
LS.CEX  = 000001   G      #6-72          10-361
LS.CIR  = 000010   G      #6-72
LS.CXO  = 010000   G      #6-72          10-389
LS.CX5  = 000400   G      #6-72
LS.DLM  = 004000   G      #6-72
LS.ECH  = 001000   G      #6-72
LS.FDX  = 004000   G      #6-72
LS.HDX  = 010000   G      #6-72
LS.LIN  = 000003   G      #6-72
LS.LMC  = 000007   G      #6-72
LS.NTI  = 004000   G      #6-72
LS.ON   = 000011   G      #6-72
LS.OPT  = 000400   G      #6-72
LS.PRO  = 000002   G      #6-72
LS.PWF  = 040000   G      #6-72
LS.TOP  = 002000   G      #6-72
LS.UNF  = 020000   G      #6-72          10-409
LS.X25  = 000012   G      #6-72
LS.11D  = 004000   G      #6-72
LX.CEX  = 000004   G      #6-72
LX.LIN  = 000006   G      #6-72
LX.PRO  = 000005   G      #6-72
```

```
                                  .IF DF  R$$MPL                 ; VNP'D SYSTEM IMAGE HACK       ;[MP01]

                         ;+
                         ; THIS ROUTINE WILL BE REMOVED FOR NEXT BASE LEVEL,AFTER A
                         ; MORE GENERAL SOLUTION IS DESIGNED .
                         ;
                         ;
                         ; $NXHCK - INIT DLX THRU POWERFAIL CALL IF SYSTEM IS VNP'D
                         ;
                         ; INPUTS: NONE
                         ;
                         ; OUTPUTS: DLX CALLED AT POWERFAIL SYSTEM IS VNP'D
                         ;+

                         $NXHCK::CMP      NTISEN+S.DATN,#^RNTL ; IS THIS FROM NTL ?
                                 BEQ      20$              ; IF YES - BRANCH  (NX POWERFAIL COMPLETED)

                                 SWSTK$   20$              ; ENTER KERNEL STATE

                                 MOV      #"NX,R3          ;* GET DEVICE NAME
                                 MOV      @DEVHD,R0        ;* GET HEAD OF DEVICE LIST
                         10$:    MOV      (R0),R0          ;* GET NEXT DEVICE
                                 BEQ      20$              ;* IF END - BRANCH
                                 CMP      R3,D.NAM(R0)     ;* IS THIS NX ?
                                 BNE      10$              ;* IF NO - BRANCH

                                 MOV      R0,R3            ;* SAVE DCB ADDRESS
                                 MOV      D.UCB(R3),R5     ;* GET UCB ADDRESS
                                 MOV      U.SCB(R5),R4     ;* GET SCB ADDRESS
                                 MOV      D.DSP(R3),R2     ;* GET DISPATCH TABLE
                                 CALL     @LDPWF           ;* CALL NX AT POWERFAIL ROUTINE

                         20$:    RETURN


                                 .ENDC


        000001                   .END
```

```
   CCCCCCCCCCCC     FFFFFFFFFFFFFFF     EEEEEEEEEEEEEEE
   CCCCCCCCCCCC     FFFFFFFFFFFFFFF     EEEEEEEEEEEEEEE
   CCCCCCCCCCCC     FFFFFFFFFFFFFFF     EEEEEEEEEEEEEEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFFFFFFFFFFF        EEEEEEEEEEEE
CCC                 FFFFFFFFFFFF        EEEEEEEEEEEE
CCC                 FFFFFFFFFFFF        EEEEEEEEEEEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
CCC                 FFF                 EEE
   CCCCCCCCCCCC     FFF                 EEEEEEEEEEEEEEE
   CCCCCCCCCCCC     FFF                 EEEEEEEEEEEEEEE
   CCCCCCCCCCCC     FFF                 EEEEEEEEEEEEEEE
```

FILEID**CFECOM

```
        .TITLE  CFEDEF - PROCESS DEFINE COMMAND
        .IDENT  /V05.00/
        .ENABL  LC
;
; COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; CFE - PROCESS DEFINE COMMAND
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
                                                                                         G 14
CFEDEF - PROCESS DEFINE COMMAND MACRO V05.03b Saturday 29-Jun-85 04:55  Page 12-1
Symbol table

CS.THI= ****** GX       OC.MBL= ****** GX       OS.BUF= ****** GX       XP.KNG= ****** GX       $SENSP= ****** GX
CS.TLO= ****** GX       OC.MWI= ****** GX       OS.PRT= ****** GX       XP.LV3= ****** GX       $SEOBJ= ****** GX
C.STS = ****** GX       OC.MXR= ****** GX       P$SNAM= ****** GX       XP.NNA= ****** GX       $SEPAR= ****** GX
DSPTBL  000000R         OC.NWC= ****** GX       P$STHI= ****** GX       $CFDEF  000252RG        $SEPRO= ****** GX
ERRIGR  000115R         OC.PVC= ****** GX       P$STLO= ****** GX       $CFERR= ****** GX       $SEPVC= ****** GX
ERRIWC  000034R         OC.RPR= ****** GX       SETCIR  000622R         $COMP = ****** GX       $SEREM= ****** GX
ERRPSI  000170R         OC.SLT= ****** GX       SETEXE  000264R         $ESUB = ****** GX       $SEROU= ****** GX
EX$ERR= 000002          OC.STA= ****** GX       SETKNC  000632R         $FLAGS= ****** GX       $SESLT= ****** GX
E2.NSP= ****** GX       OC.SVC= ****** GX       SETLIN  000536R         $NNAM = ****** GX       $SESNA= ****** GX
E2.ROU= ****** GX       OC.TRI= ****** GX       SETPVC  001110R         $OPRND= ****** GX       $SESVC= ****** GX
FL.CRE= ****** GX       OC.UNT= ****** GX       SETSVC  001116R         $OPRN2= ****** GX       $SEUNT= ****** GX
FL.PVC= ****** GX       OE.NOD= ****** GX       SETSYS  001174R         $PSIPT= ****** GX       $SEXAC= ****** GX
FL.X25= ****** GX       OE.NSP= ****** GX       SETTRN  001030R         $SEBUF= ****** GX       $SEXSV= ****** GX
OC.APR= ****** GX       OE.ROU= ****** GX       SETXPR  001226R         $SECNT= ****** GX       $SEX2P= ****** GX
OC.CNT= ****** GX       OE.SUB= ****** GX       SW.SLT= ****** GX       $SECUG= ****** GX       $SEX29= ****** GX
OC.CST= ****** GX       OL.CNT= ****** GX       WC.CIR= ****** GX       $SEDTE= ****** GX       $SEX3P= ****** GX
OC.CTM= ****** GX       OL.SLT= ****** GX       XP.CUG= ****** GX       $SELOG= ****** GX       $SWICH= ****** GX
OC.DTE= ****** GX       OL.UNT= ****** GX       XP.DTE= ****** GX       $SENOD= ****** GX       $WILD = ****** GX
OC.HTM= ****** GX       CL.X25= ****** GX       XP.KND= ****** GX

. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        001342    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8544 Words  ( 34 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:11.57
SY:CFEDEF.V2,[132,134]CFEDEF/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]CFEDEF
```

```
                                                                          H 14
CFEDEF      CREATED BY  MACRO  ON 29-JUN-85 AT 04:55      PAGE 1
```

```
                                        .TITLE  CFEHLP - HELP ROUTINE FOR CFE
                                        .IDENT  /V05.00/

                                ;
                                ; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ; CFE - HANDLE PROCESSING OF HELP INFORMATION
                                ;
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  14-DEC-79
                                ;       DECNET-11M/S V3.0
                                ;       DECNET-11M-PLUS V1.0
                                ;
                                ; 3.00  16-APR-82
                                ;       DECNET-11M V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;
```

```
                                     ;
001532                               $PRNAM::.BLKW    1                ; STORAGE FOR THE DEVICE/PROCESS NAME (RAD50)
               001532'                       $DVNAM == $PRNAM          ; ALIAS FOR DEVICE NAME
001534                               $CNTNM::.BLKW    1                ; THE CONTROLLER NUMBER
001536                               $UNTNM::.BLKW    1                ; THE UNIT NUMBER
001540                               $STAN:: .BLKW    1                ; THE STATION NUMBER
                                     ;
                                     ; THE PREVIOUS LINE MUST BE CONTIGUOUS
                                     ;
001542                               $NODNA::.BLKW    2                ; NODE NAME (2 WORDS RAD50)
001546                               $NODAD::.BLKW    1                ; NODE ADDRESS
001550                               $VEC::  .BLKW    1                ; THE VECTOR ADDRESS
001552                               $PRI::  .BLKW    1                ; THE PRIORITY
001554                               $BAUD:: .BLKW    1                ; THE BAUD RATE
001556                               $CSR::  .BLKW    1                ; CSR ADDRESS
001560                               $SCSR:: .BLKW    1                ; SECONDARY CSR ADDRESS
001562                               $LINKS::.BLKW    1                ; MAXIMUM NUMBER OF LOGICAL LINKS
001564                               $POLNA::.BLKW    2                ; POOL NAME (2 WORDS RAD50)
001570                               $PARNA::.BLKW    2                ; PARTITION NAME (2 WORDS RAD50)

001574                               $COPES::                          ; NUMBER OF OBJECT TASK COPIES  (OBJECT)
001574                               $RST::                            ; RESTART TIMER                 (X25-PRO)
001574                                       .BLKW    1
001576                               $TNAM:: .BLKW    2                ; OBJECT TASK NAME (RAD50)
001602                               $TNUM:: .BLKW    1                ; OBJECT TASK NUMBER
001604                               $EVENT::.BLKW    4                ; EVENT FILTER MASKS
001614                               $CLASS::.BLKW    1                ; EVENT CLASS
001616                               $EVBEG::.BLKW    1                ; BEGINNING EVENT TYPE IN EVENT RANGE
001620                               $VLVL:: .BLKW    1                ; OBJECT TASK VERIFICATION LEVEL
001622                               $DTE::
001622                               $ID::   .BLKW    1                ; NODE ID LENGTH
001624                                       .BLKB    32.              ; 32 CHAR ASCII NODE ID STRING
001664                                       .BLKW    33.-<.-$ID>+12   ; 32 WORD CHANNEL LIST
               001634'                       $CHNLS == $ID + 12        ; CHANNEL LIST - LEAVE ROOM FOR DTE STRING
               001704'                       $CHNND == .-2
               000050                        $CHNLN == .-$CHNLS-2      ; LENGTH OF CHANNEL LIST
001706                               $NLN::  .BLKW    1                ; NUMBER OF CHANNELS IN THE SYSTEM
001710                               $COST:: .BLKW    1                ; COST OF A LINE
001712                               $OBJ::  .BLKW    2                ; OBJECT ID            (DESTINATION)
               001550'                       $MAXC == $VEC             ; MAXIMUM ROUTING COST
               001552'                       $MAXH == $PRI             ; MAXIMUM ROUTING HOPS
               001556'                       $NN   == $CSR             ; MAXIMUM NUMBER OF NODES IN NETWORK
               001574'                       $NCOU == $COPES           ; MAXIMUM NUMBER OF NODE COUNTERS FOR NSP
               001564'                       $NLT  == $POLNA           ; MAXIMUM NUMBER OF LINE TABLES IN PROCESS SPACE
               001620'                       $NCT  == $VLVL            ; MAXIMUM NUMBER OF CONTROLLER TABLFS FOR PROCESS
               001550'                       $BYT  == $VEC             ; NTPOOL BYTE-AREA EXTENSION
               001566'                       $STADR  == $POLNA+2       ; STATION ADDRESS
               001562'                       $URM    == $LINKS         ; UNIBUS RUN MASK (FOR RSX-11M+)
               001450'                       $CMDPT  == $COMP          ; POINTER TO LINE AFTER ''HELP''
               001570'                       $DPR    == $PARNA         ; DEAD POLLING RATIO
               001572'                       $APR    == $PARNAM+2      ; ACTIVE POLLING RATIO
               001560'                       $HOST   == $SCSR          ; HOST NODE ADDRESS
               001712'                       $CIRID  == $OBJ           ; CIRCUIT ID (PVC OR SVC)
001716                               $DST::  .BLKW    2                ; DESTINATION ID       (MODULE)
               001716'                       $CIMXR  == $DST           ; MAXIMUM ROUTERS      (CIRCUIT)
               001720'                       $CIRPR  == $DST+2         ; ROUTER PRIORITY      (CIRCUIT)
               001716'                       $ESUB   == $DST           ; EXECUTOR SUBADDRESSES (EXEC NODE, PSI)
```

```
NTIUMR  MACRO V05.03b Saturday 29-Jun-85 03:28  Page 7-1
Symbol table

A$$CHK= 000000          G$$WRD= 000000          M$$OVR= 000000          P$$P45= 000000          S.CTM   000006
A$$CPS= 000000          I$$RAR= 000000          M.BFVH  000011          P$$WRD= 000000          S.DMCS  000020
A$$PRI= 000000          I$$RDN= 000000          M.BFVL  000012          Q$$OPT= 000010          S.FRK   000016
A$$TRP= 000000          K$$CNT= 177546          M.LGTH= 000014          R$$DER= 000000          S.ITM   000007
C$$CKP= 000000          K$$CSR= 177546          M.LNK   000000          R$$K11= 000001          S.LHD   000000
C$$ORE= 000400          K$$LDC= 000000          M.UMRA  000002          R$$SND= 000000          S.PKT   000014
C$$RSH= 177564          K$$TPS= 000074          M.UMRN  000004          R$$11M= 000000          S.PRI   000004
D$$BUG= 177514          LD$LP = 000000          M.UMVH  000010          SPARE = 000010          S.RCNT  177772
D$$ISK= 000000          L$$ASG= 000000          M.UMVL  000006          SP.EIP= 000001          S.ROFF  177773
D$$L11= 000001          L$$DRV= 000000          N$$ACC= 000001          SP.ENB= 000002          S.STS   000011
D$$YNC= 000000          L$$P11= 000001          N$$BUF= 000001          SP.LOG= 000004          S.VCT   000005
D$$YNM= 000000          L$$11R= 000000          N$$LDV= 000001          S$$WRG= 000000          T$$KMG= 000000
E$$XPR= 000000          M$$CRB= 000124          N$$MCP= 000001          S$$YSZ= 007600          T$$MIN= 000000
F$$LVL= 000001          M$$CRX= 000000          N$$MLL= 000001          S.BMSK  177776          V$$CTR= 001000
G$$TPP= 000000          M$$FCS= 000000          N$$MOV= 000010          S.BMSV  177774          X$$DBT= 000000
G$$TSS= 000000          M$$MGE= 000000          N$$NCT= 000001          S.CON   000010          $NIUMR  000000RG
G$$TTK= 000000          M$$NET= 000000          N$$PEM= 000001          S.CSR   000012

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000056    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 9373 Words  ( 37 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.19
SY:NTIUMR.V2,[132,134]NTIUMR/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]NTIUMR
```

```
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE                REFERENCES

ZF.DVP  = 100000              #5-60
ZF.INI  = 040000              #5-60
ZF.KMX  = 000020              #5-60
ZF.LLC  = 000004              #5-60
ZF.LMC  = 000100              #5-60
ZF.MAN  = 020000              #5-60
ZF.MFL  = 000010              #5-60
ZF.MTM  = 000400              #5-60
ZF.MUX  = 000040              #5-60
ZF.PSE  = 002000              #5-60
ZF.SLI  = 010000              #5-60
ZF.TIM  = 000200              #5-60
ZF.X3P  = 000000              #5-60
ZS.ASN  = 100000              #5-60
ZS.BSY  = 140000              #5-60
Z.AVL     000014              #5-60
Z.DAT     000016              #5-60
Z.DSP     000000              #5-60        5-60
Z.FLG     000010              #5-60
Z.LEN   = 000016              #5-60
Z.LLN     000006              #5-60
Z.MAP     000020              #5-60
Z.NAM     000004              #5-60
Z.PCB     000012              #5-60
Z.SCH     000007              #5-60
$ERR40    000036 R            #6-95        7-187
$ERR41    000076 R            #6-96        7-188
$ERR42    000136 R            #6-97        7-189
$HEADR  = ******   GX          7-165
$NIERR  = ******   GX          6-95        6-96         6-97
$NIMFE  = ******   GX          7-171
$NIX25    000000 RG           #7-124
$$$GLB  = ******               6-88        6-89
$$$T1   = 000002              #6-88        6-88         6-88         6-88        #6-89
$$$T2   = 000032 R            #6-89        6-89
.BASEB  = 140000              #5-59
```

```
                                                                 H  3
NTICEX  MACRO V05.03b Saturday 29-Jun-85 03:29  Page 9-2
Symbol table

PDVNM = ****** GX       R$$DER= 000000          S.COST  000001          ZF.DLC= 000002          Z.DAT   000016
PDVTA = ****** GX       R$$K11= 000001          S.FLG   000000          ZF.DVP= 100000          Z.DSP   000000
PIRQ  = 177772          R$$SND= 000000          S.LEN   000004          ZF.INI= 040000          Z.FLG   000010
PMODE = 030000          R$$11M= 000000          S.NMST  000002          ZF.KMX= 000020          Z.LEN = 000016
PR0   = 000000          R$$11S= 000001          S.OWNR  000003          ZF.LLC= 000004          Z.LLN   000006
PR1   = 000040          SF.ACT= 000200          TPS   = 177564          ZF.LMC= 000100          Z.MAP   000020
PR2   = 000100          SF.ENA= 000100          T$$KMG= 000000          ZF.MAN= 020000          Z.NAM   000004
PR3   = 000140          SF.LPB= 000004          T$$MIN= 000000          ZF.MFL= 000010          Z.PCB   000012
PR4   = 000200          SF.MFL= 000040          UBMPR = 170200          ZF.MTM= 000400          Z.SCH   000007
PR5   = 000240          SF.PAC= 000020          UISAR0= 177640          ZF.MUX= 000040          $NICEX  000000RG
PR6   = 000300          SF.REA= 000010          UISAR1= 177642          ZF.PSE= 002000          $NICSR= ****** GX
PR7   = 000340          SF.SER= 000001          UMRPT = ****** GX       ZF.SLI= 010000          $NIDLI= ****** GX
PS    = 177776          SF.SVC= 000002          V$$CTR= 001000          ZF.TIM= 000200          $NITIM= ****** GX
PUMR  = ****** GX       SF.UNL= 000040          X$$DBT= 000000          ZF.X3P= 000000          $NIUMR= ****** GX
P$$P45= 000000          SWR   = 177570          ZF.COU= 001000          ZS.ASN= 100000          .BASEB= 140000
P$$WRD= 000000          S$$WRG= 000000          ZF.DDM= 000001          ZS.BSY= 140000          .CXCSR= ****** GX
Q$$OPT= 000010          S$$YSZ= 007600          ZF.D'A= 004000          Z.AVL   000014          .$$$$.= 000034


. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000212    001   (RW,I,LCL,REL,CON)
DATA    000002    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 49
Work  file writes: 60
Size of work file: 20950 Words  ( 82 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:19.49
SY:NTICEX11S.V2,[132,134]NTICEX11S/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]V2,NTICEX
```

```
                                                                 I  3
NTICEX11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 1
```

```
                                                              H  4
NTICSR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE             REFERENCES

L.NMST     000020          #6-62
L.NSTA     000014          #6-62
L.OWNR     000021          #6-62
L.UNT      000013          #6-62
MPAR     = 172100          #6-61
MPCSR    = 177746          #6-61
M$$MGE   = 000000           6-61
PDVTA    = ******    GX     7-163
PIRQ     = 177772          #6-61
PMODE    = 030000          #6-61
PR0      = 000000          #6-61
PR1      = 000040          #6-61
PR2      = 000100          #6-61
PR3      = 000140          #6-61
PR4      = 000200          #6-61
PR5      = 000240          #6-61
PR6      = 000300          #6-61
PR7      = 000340          #6-61
PS       = 177776          #6-61
R$$MPL   = ******           6-61          6-61          6-61
R$$11D   = ******           6-61          6-63
R$$11M   = 000000           6-63
R$$11S   = 000001          #4-2           6-63          7-169
SF.ACT   = 000200          #6-62
SF.ENA   = 000100          #6-62
SF.LPB   = 000004          #6-62
SF.MFL   = 000040          #6-62
SF.PAC   = 000020          #6-62
SF.REA   = 000010          #6-62
SF.SER   = 000001          #6-62
SF.SVC   = 000002          #6-62
SF.UNL   = 000040          #6-62
SWR      = 177570          #6-61
S$$BAS   = ******           6-88          6-88          7-159         7-183
S.COST     000001          #6-62
S.FLG      000000          #6-62
S.LEN      000004          #6-62
S.NMST     000002          #6-62
S.OWNR     000003          #6-62
S.VTTA   = 000002           6-74          6-75
TPS      = 177564          #6-61
UBMPR    = 170200          #6-61
UISAR0   = 177640          #6-61
UISAR1   = 177642          #6-61
X$$MCB   = ******           6-63          6-63
ZF.COU   = 001000          #6-63
ZF.DDM   = 000001          #6-63
ZF.DIA   = 004000          #6-63
ZF.DLC   = 000002          #6-63
ZF.DVP   = 100000          #6-63
ZF.INI   = 040000          #6-63
ZF.KMX   = 000020          #6-63
```

```
                                                              I  4
NTICSR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 3
```

Table of contents

```
     6-   54     MACROS
     7-   81     LOCAL DATA
     8-  107     ERROR MESSAGES
     9-  124     SET EXECUTOR ON
    10-  300     FNDPLI - GET PLI'S PDV INDEX
```

```
Symbol table

        000404    001   (RW,I,LCL,REL,CON)
DATA    000366    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 123
Work  file writes: 126
Size of work file: 29428 Words  ( 115 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:33.21
SY:NTILON11S.V2,[132,134]NTILON11S/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]V2,NTILON
```

```
                .SBTTL  DLMSET - SET UP COUNTER BLOCK FIELDS FOR DLM CIRCUITS

        ;+
        ;
        ; DLMSET - SET UP COUNTER BLOCK FIELDS FOR DLM CIRCUITS
        ;
        ; INPUTS:
        ;       STAADD - ADDRESS OF STATION TABLE
        ;       R0 - XPT'S PDV ADDRESS
        ;       R2 - ADDRESS OF REVERSE MAPPING TABLE ENTRY
        ;
        ; OUTPUTS:
        ;       DLM FLAGS AND SVC DESCRIPTOR ADDRESS OR PORT NUMBER (PVC) STORED
        ;          IN COUNTER BLOCK
        ;       COUNTER BLOCK IS ALLOCATED IF IT DOES NOT ALREADY EXIST
        ;
        ;-
                .IF NDF R$$11S
        DLMSET:
                SWSTK$  60$                     ;; ENTER SYSTEM STATE
                MOV     (R2),R4                 ;; GET CHANNEL NUMBER AND PDV
                MOV     @PSIPT,R2               ;; GET POINTER TO PSI HOME BLOCK
                BNE     10$                     ;; BR IF HOME BLOCK EXISTS
                RETC    R2                      ;; ELSE SET USER C-BIT
                BR      60$                     ;; AND EXIT
        10$:    MOV     R0,R3                   ;; POINT TO XPT'S PDV
                ADD     #Z.MAP,R3               ;; POINT TO FORWARD MAPPING
                CLR     -(SP)                   ;; GET CHANNEL NUMBER
                MOVB    R4,(SP)                 ;; ...
                ASL     (SP)                    ;; MAKE IT A WORD INDEX
                ADD     (SP)+,R3                ;; POINT TO SLN AND STATION NUMBER
                MOV     (R3),R4                 ;; GET SLN AND STATION NUMBER
                SAVMAP                          ;; SAVE CURRENT MAPPING
                MOV     Z.DAT(R0),R0            ;; GET ADDRESS OF XPT'S DDB
                MAP     N$TLC+2(R0)             ;; MAP TO CIRCUIT COUNTERS
                MOV     N$TLC(R0),R3            ;; GET NUMBER OF COUNTERS
                MOV     N$TLC+4(R0),R0          ;; GET ADDRESS OF FIRST COUNTER BLOCK
                BIAS    R0                      ;; BIAS COUNTER BLOCK ADDRESS
        20$:    TST     (R0)                    ;; HAS THIS BLOCK BEEN ASSIGNED ALREADY?
                BMI     30$                     ;; BR IF NO - USE IT
                CMP     R4,(R0)                 ;; ELSE IS THIS THE CORRECT ENTRY?
                BEQ     30$                     ;; BR IF YES
                ADD     #T$LLEN,R0              ;; POINT TO NEXT COUNTER BLOCK
                SOB     R3,20$                  ;; CHECK ALL COUNTER BLOCKS
                BR      50$                     ;; EXIT
        30$:    BIS     #PX$DLM,T$FLAG(R0) ;; SET FLAG TO INDICATE DLM
                BITB    #SF.SVC,@STAADD ;; IS THIS AN SVC?
                BEQ     50$                     ;; BR IF NO
                BIS     #PX$SVC,T$FLAG(R0) ;; ELSE INDICATE SVC
                SAVMAP                          ;; SAVE CURRENT MAPPING
                ADD     #H$SVC,R2               ;; GET FIRST SVC DESCRIPTOR ADDRESS
        40$:    MOV     (R2),R2                 ;; GET NEXT BLOCK IN LIST
                BEQ     45$                     ;; BR IF NOT FOUND
                MOV     R2,-(SP)                ;; GET UNMAPPED ADDRESS
                CALL    @CEACC                  ;; CONVERT TO MAPPED ADDRESS
                MOV     (SP)+,R2                ;; RETRIEVE MAPPED ADDRESS
                CMPB    S$CTL(R2),CTLNUM;; IS THIS THE CORRECT SVC DESCRIPTOR?
```

```
     54                                     .SBTTL  MACRO DEFINITIONS
     55
     56                             ;
     57                             ; MACRO CALLS
     58                             ;
     59                                     .MCALL  EMSG$,NTLER$,QIOW$S,EXIT$S,ASTX$S,WTSE$S,WSIG$S,CLEF$S
     60                                     .MCALL  DIR$,CLEF$S,SETF$S,SRDA$S,RCVD$,SDAT$,RQST$,DSAR$S,ENAR$S
     61                                     .MCALL  CCBDF$,NHWDF$,NTLDF$,PDVDF$,SLTDF$,OPTDF$,MAP,CALLX,CALLR
     62                                     .MCALL  SDAT$S,EMSG$R,RCVX$,RETC,MRKT$,VECDF$
     63                                     .MCALL  SCBDF$,PCBDF$,TCBDF$,SAVMAP,RESMAP
     64
     65                                     .IF DF  R$$MPL
     66                                     .MCALL  GIN$
     67                                     .ENDC
     68
     69                             ;
     70                             ; CONSTANTS AND OFFSETS
     71                             ;
     72 000000                              NTLDF$  <:>,<=>,NTLSYM          ;NTL REQ. BLOCK AND SYMBOLS (GLOBALLY)
     73 000000                              CCBDF$                          ;CCB OFFSETS
     74 000000                              OPTDF$                          ;COMM/EXEC OPTIONS WORD
     75 000000                              PDVDF$                          ;PDV OFFSETS
     76 000000                              SLTDF$                          ;SLT OFFSETS
     77 000000                              NHWDF$                          ;HARDWARE OFFSETS
     78 000000                              PCBDF$                          ;PARTITION CONTROL BLOCK OFFSETS
     79 000000                              SCBDF$                          ;UMR BLOCK O,D
     80 000000                              TCBDF$                          ;TCB OFFSETS
     81 000000                              VECDF$                          ;VECTOR BLOCK OFFSETS
     82
```

```
                                          .SBTTL  FIND PDV INDICES OF NSP PROCESSES (ECL AND XPT)
                                  ;+
                                  ; FNDPDV - FIND PDV INDICES OF NSP PROCESSES (ECL AND XPT)
                                  ;
                                  ; SCAN THE PDV INDEX TABLE FOR PROCESSES WITH THE NAMES 'ECL' AND 'XPT'
                                  ; AND SAVE THEIR RESPECTIVE PDV INDICES AND DATA BASE ADDRESSES.
                                  ;
                                  ; INPUTS:
                                  ;       NONE
                                  ;
                                  ; OUTPUTS:
                                  ;       $NIECL = ECL PDV INDEX OR ZERO
                                  ;       $DBECL = ADDRESS OF ECL'S DDB OR ZERO
                                  ;       $NIXPT = XPT PDV INDEX OR ZERO
                                  ;       $DBXPT = ADDRESS OF XPT'S DDB OR ZERO
                                  ;-

000204  005067  000000'           FNDPDV::CLR     $NIECL                ;ASSUME ECL AND XPT
000210  005067  000002'                   CLR     $NIXPT                ;  ARE NOT IN THE SYSTEM
000214  005067  000004'                   CLR     $DBECL                ;   ...
000220  005067  000006'                   CLR     $DBXPT                ;   ...

000224                                    SWSTK$  20$                   ; ENTER SYSTEM STATE
000230  012702  017704                    MOV     #^RECL,R2             ;;LOOK FOR ECL'S PDV
000234                                    CALL    @PDVID                ;;  ...
000240  103413                            BCS     10$                   ;;IF CS, NOT FOUND
000242  010267  000000'                   MOV     R2,$NIECL             ;;SAVE ECL'S PDV INDEX
000246  067702  000530'                   ADD     @PDVTA,R2             ;;POINT INTO PDV INDEX TABLE
000252  011202                            MOV     (R2),R2               ;;GET ADDRESS OF ECL'S PDV
000254  005762  000012                    TST     Z.PCB(R2)             ;;IS THE PROCESS LOADED?
000260  001403                            BEQ     10$                   ;;IF EQ, NO
000262  016267  000016  000004'           MOV     Z.DAT(R2),$DBECL      ;;SAVE ADDRESS OF ECL'S DATABASE

000270  012702  114224            10$:    MOV     #^RXPT,R2             ;;LOOK FOR XPT'S PDV
000274                                    CALL    @PDVID                ;;  ...
000300  103413                            BCS     20$                   ;;IF CS, NOT FOUND
000302  010267  000002'                   MOV     R2,$NIXPT             ;;SAVE XPT'S PDV INDEX
000306  067702  000530'                   ADD     @PDVTA,R2             ;;POINT INTO PDV INDEX TABLE
000312  011202                            MOV     (R2),R2               ;;GET ADDRESS OF XPT'S PDV
000314  005762  000012                    TST     Z.PCB(R2)             ;;IS THE PROCESS LOADED?
000320  001403                            BEQ     20$                   ;;IF EQ, NO
000322  016267  000016  000006'           MOV     Z.DAT(R2),$DBXPT      ;;SAVE ADDRESS OF XPT'S DATABASE

000330                            20$:    RETURN
```

```
                                                                  H 10
NTINIT11S  CREATED BY  MACRO  ON 5-SEP-85 AT 11:10        PAGE 3
SYMBOL  CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE               REFERENCES

L.COST      000015           #6-76
L.CTL       000012           #6-76
L.CVA       177776           #6-76
L.DDM       000002           #6-76
L.DDS       000004           #6-76
L.DLC       000003           #6-76
L.DLM       000006           #6-76
L.DLS       000010           #6-76
L.FLG       000000           #6-76
L.KRBA      000016           #6-76
L.LEN     = 000022           #6-76
L.MPF       000022           #6-76
L.NMST      000020           #6-76
L.NSTA      000014           #6-76
L.OWNR      000021           #6-76
L.UNT       000013           #6-76
MPAR      = 172100           #6-77
MPCSR     = 177746           #6-77
MRKTEF    = 000001           #7-108         7-109          10-402
M$$MGE    = 000000            6-77
M.KTEF    = 000002            7-109
NTIERB      000137 R         #8-165         14-669
NTIERN      000124 R         #8-164         14-688         14-689
NTIGRQ      000006 R         #10-355
NTINIT      000000 R         #9-284         18-881
NTIRCX      000016 R         #7-104         9-309
NTLPT       000470 RG        #8-234
PDVID       000522 RG        #8-247         13-592         13-602
PDVNM       000520 RG        #8-246
PDVTA       000530 RG        #8-250         13-595         13-605
PIRQ      = 177772           #6-77
PMODE     = 030000           #6-77
PR0       = 000000           #6-77
PR1       = 000040           #6-77
PR2       = 000100           #6-77
PR3       = 000140           #6-77
PR4       = 000200           #6-77
PR5       = 000240           #6-77
PR6       = 000300           #6-77
PR7       = 000340           #6-77
PS        = 177776           #6-77
PSIPT       000506 RG        #8-241
PUMR        000442 RG        #8-218
QASTT       000450 RG        #8-221
R$$MPL    = ******            6-65          6-77           6-77           6-77           7-90           8-206          8-228          9-285          11-481
R$$11D    = ******            6-75          6-77
R$$11M    = 000000            6-75
R$$11S    = 000001           #4-2           6-75           7-115          8-189          9-297          10-356         10-364         10-392         10-408
                              10-422        10-438         12-554         16-760         17-794         18-847
R.QSTN    = 000002            7-110
R.VXTN    = 000002            7-104
SF.ACT    = 000200           #6-76

                                                                  I 10
NTINIT11S  CREATED BY  MACRO  ON 5-SEP-85 AT 11:10        PAGE 4
```

```
NTITIM  MACRO V05.03b Saturday 29-Jun-85 03:33  Page 10-1
Symbol table

ABTIM = ****** GX        C.SRC   000014          K$$CNT= 177546          N$$NCT= 000001          S$$WRG= 000000
A$$CHK= 000000           C.SSHT= 000004          K$$CSR= 177546          N$$PEM= 000001          S$$YSZ= 007600
A$$CPS= 000000           C.SUB   000012          K$$LDC= 000000          PIRQ   = 177772         TIFLG   000000R
A$$PRI= 000000           C.SYST= 000006          K$$TPS= 000074          PMODE  = 030000         TKPS   = ****** GX
A$$TRP= 000000           C.SYTK= 000010          LD$LP  = 000000         PR0    = 000000         TK100  = ****** GX
BUFUMP= 172354           C.TCB   000004          L$$ASG= 000000          PR1    = 000040         TPS    = 177564
CLINS   000162R          C.TIM   000006          L$$DRV= 000000          PR2    = 000100         T$$KMG= 000000
CLKHD = ****** GX        C.UIC   000016          L$$P11= 000001          PR3    = 000140         T$$MIN= 000000
CMODE = 140000           D$$BUG= 177514          L$$11R= 000000          PR4    = 000200         T1     = 000002
C$$CKP= 000000           D$$ISK= 000000          MPAR   = 172100         PR5    = 000240         T1SCL  = ****** GX
C$$ORE= 000400           D$$L11= 000001          MPCSR  = 177746         PR6    = 000300         T1SIN    000116R
C$$RSH= 177564           D$$YNC= 000000          M$$CRB= 000124          PR7    = 000340         T100C  = ****** GX
C.AR5   000014           D$$YNM= 000000          M$$CRX= 000000          PS     = 177776         T100Q  = ****** GX
C.AST   000012           E$$XPR= 000000          M$$FCS= 000000          P$$P45= 000000          T5     = 000001
C.CSTP= 000012           F$$LVL= 000001          M$$MGE= 000000          P$$WRD= 000000          T50IN    000130R
C.DST   000016           G$$TPP= 000000          M$$NET= 000000          Q$$OPT= 000010          UBMPR  = 170200
C.EFN   000003           G$$TSS= 000000          M$$OVR= 000000          R$$DER= 000000          UISAR0= 177640
C.LGTH= 000020           G$$TTK= 000000          N$$ACC= 000001          R$$K11= 000001          UISAR1= 177642
C.LNK   000000           G$$WRD= 000000          N$$BUF= 000001          R$$SND= 000000          V$$CTR= 001000
C.MRKT= 000000           I$$RAR= 000000          N$$LDV= 000001          R$$11M= 000000          X$$DBT= 000000
C.RQT   000002           I$$RDN= 000000          N$$MCP= 000001          R$$11S= 000001          $NITIM   000002RG
C.RSI   000012           KISAR0= 172340          N$$MLL= 000001          SWR    = 177570         .BASEB= 140000
C.SCHD= 000002           KISAR6= 172354          N$$MOV= 000010

. ABS.  000020    000   (RW,I,GBL,ABS,OVR)
        000250    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10026 Words  ( 40 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:09.38
SY:NTITIM11S.V2,[132,134]NTITIM11S/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]V2,NTITIM
```

```
NTITIM11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:33      PAGE 1
```

```
**FILE**ID**CFECOM
```

```
  CCCCCCCC   FFFFFFFFFF   EEEEEEEEEE     CCCCCCCC     OOOOOO     MM      MM
  CCCCCCCC   FFFFFFFFFF   EEEEEEEEEE     CCCCCCCC     OOOOOO     MM      MM
CC           FF           EE           CC           OO      OO   MMMM  MMMM
CC           FF           EE           CC           OO      OO   MMMM  MMMM
CC           FF           EE           CC           OO      OO   MM  MM  MM
CC           FF           EE           CC           OO      OO   MM  MM  MM
CC           FFFFFFFF     EEEEEEEE     CC           OO      OO   MM      MM
CC           FFFFFFFF     EEEEEEEE     CC           OO      OO   MM      MM
CC           FF           EE           CC           OO      OO   MM      MM
CC           FF           EE           CC           OO      OO   MM      MM
CC           FF           EE           CC           OO      OO   MM      MM
CC           FF           EE           CC           OO      OO   MM      MM        ....
  CCCCCCCC   FF           EEEEEEEEEE     CCCCCCCC     OOOOOO     MM      MM        ....
  CCCCCCCC   FF           EEEEEEEEEE     CCCCCCCC     OOOOOO     MM      MM        ....


LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                    ;****
                                    ; LOCAL MACROS
                                    ;****

                                            .MACRO  PRINT   TEXT
                                            MOV     TEXT,R0                 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
                                            CALL    $PRINT                  ; PRINT MESSAGE
                                            .ENDM   PRINT

                                            .MACRO  ERROR$  TEXT
                                    ;
                                    ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
                                    ; MESSAGE STRING.
                                    ;
                                            .IF DIF <TEXT><R0>
                                            MOV     TEXT,R0                         ; GET ADDRESS OF ERROR MESSAGE
                                            .ENDC
                                            CALL    $CFERR                          ; PRINT ERROR MESSAGE
                                            .ENDM   ERROR$

                                            .MACRO  EPRINT  TEXT
                                    ;
                                    ; PRINT TEXT ON ERROR LUN
                                    ;
                                            .IF DIF <TEXT><R0>
                                            MOV     TEXT,R0                         ; GET ADDRESS OF ASCIZ STRING TO PRINT
                                            .ENDC
                                            CALL    $CFER1                          ; PRINT MESSAGE ON ERROR LUN
                                            .ENDM   EPRINT

                                            .MCALL  CALLR



                                    ;****
                                    ; LOCAL DATA
                                    ;****


                                    ;
                                    ; DEFINE THE DISPATCH TABLE
                                    ;
000000                              DSPTBL:
000000  000000G                             .WORD   $SEREM                  ; ''DEFINE NODE'' REQUEST
000002  000000G                             .WORD   $SEPRO                  ; ''DEFINE PROCESS'' REQUEST
000004  000264'                             .WORD   SETEXE                  ; ''DEFINE EXECUTOR'' REQUEST
000006  001174'                             .WORD   SETSYS                  ; ''DEFINE SYSTEM'' REQUEST
000010  000000G                             .WORD   $SEOBJ                  ; ''DEFINE OBJECT'' REQUEST
000012  000536'                             .WORD   SETLIN                  ; ''DEFINE LINE'' REQUEST
000014  000000G                             .WORD   $SELOG                  ; ''DEFINE LOGGING'' REQUEST
000016  000622'                             .WORD   SETCIR                  ; ''DEFINE CIRCUIT'' REQUEST
000020  000632'                             .WORD   SETKNC                  ; ''DEFINE KNOWN CIRCUITS'' REQUEST
000022  001226'                             .WORD   SETXPR                  ; ''DEFINE MODULE X25-PROTOCOL''
000024  000000G                             .WORD   $SEXSV                  ; ''DEFINE MODULE X25-SERVER''
000026  000000G                             .WORD   $SEXAC                  ; ''DEFINE MODULE X25-ACCESS''
000030  000000G                             .WORD   $SEX29                  ; ''DEFINE MODULE X29-SERVER''
000032  000000G                             .WORD   $SESNA                  ; ''DEFINE MODULE SNA-PROTOCOL''

```

```
                                                                                 H 14
CFEDEF     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE              REFERENCES

CS.THI  = ******  GX        6-185       6-193       6-206       6-222
CS.TLO  = ******  GX        6-185       6-195       6-206       6-218
C.STS   = ******  GX        6-184      *6-206      *6-218      *6-222
DSPTBL    000000 R         #4-94        5-137
ERRIGR    000115 R         #4-116       8-324
ERRIWC    000034 R         #4-115       8-321
ERRPSI    000170 R         #4-117       6-230       8-327       12-448
EX$ERR  = 000002           #4-113       4-115       4-116       4-117
E2.NSP  = ******  GX        6-161
E2.ROU  = ******  GX        6-154
FL.CRE  = ******  GX        8-302
FL.PVC  = ******  GX        8-304       8-307
FL.X25  = ******  GX        8-298
OC.APR  = ******  GX        8-290       8-305       8-309
OC.CNT  = ******  GX        9-350
OC.CST  = ******  GX        8-290       10-386
OC.CTM  = ******  GX        8-296
OC.DTE  = ******  GX        8-296
OC.HTM  = ******  GX        8-290       10-386      10-390
OC.MBL  = ******  GX        8-296
OC.MWI  = ******  GX        8-296
OC.MXR  = ******  GX        8-290
OC.NWC  = ******  GX        8-282
OC.PVC  = ******  GX        8-296       8-309
OC.RPR  = ******  GX        8-290
OC.SLT  = ******  GX        9-347
OC.STA  = ******  GX        8-290       10-386      10-390
OC.SVC  = ******  GX        8-296       8-305
OC.TRI  = ******  GX        8-305       8-309
OC.UNT  = ******  GX        9-354       10-387
OE.NOD  = ******  GX        6-165
OE.NSP  = ******  GX        6-159
OE.ROU  = ******  GX        6-152
OE.SUB  = ******  GX        6-169
OL.CNT  = ******  GX        7-249
OL.SLT  = ******  GX        7-246
OL.UNT  = ******  GX        7-253
OL.X25  = ******  GX        7-256
OS.BUF  = ******  GX        11-415
OS.PRT  = ******  GX        11-412
P$SNAM  = ******  GX       *12-435     *12-436
P$STHI  = ******  GX       *6-221
P$STLO  = ******  GX        6-199      *6-217
SETCIR    000622 R          4-102      #8-276
SETEXE    000264 R          4-97       #6-151
SETKNC    000632 R          4-103      #8-281
SETLIN    000536 R          4-100      #7-245
SETPVC    001110 R          8-284       8-306      #10-375
SETSVC    001116 R          8-285       8-310      #10-377
SETSYS    001174 R          4-98       #11-411
SETTRN    001030 R          8-291       8-297      #9-344
SETXPR    001226 R          4-104      #12-430
```

```
                                                                                 I 14
CFEDEF     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 2
```

```
     55                                         ;****
     56                                         ; LOCAL MACROS
     57                                         ;****
     58
     59                                                 .MACRO  PRINT   TEXT
     60                                                 MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
     61                                                 CALL    $PRINT          ; PRINT MESSAGE
     62                                                 .ENDM   PRINT
     63
     64                                                 .MACRO  ERROR$  TEXT
     65                                         ;
     66                                         ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
     67                                         ; MESSAGE STRING.
     68                                         ;
     69                                                 .IF DIF <TEXT><R0>
     70                                                 MOV     TEXT,R0                 ; GET ADDRESS OF ERROR MESSAGE
     71                                                 .ENDC
     72                                                 CALL    $CFERR                  ; PRINT ERROR MESSAGE
     73                                                 .ENDM   ERROR$
     74
     75                                         ;****
     76                                         ; MACRO CALLS
     77                                         ;****
     78
     79                                                 .MCALL  DIR$,HLPDF$,CALLR
     80
     81                                         ;****
     82                                         ; LOCAL DATA
     83                                         ;****
     84
     85                                         ;
     86                                         ; HELP FILE DEFINITION
     87                                         ;
     88 000000                                          HLPDF$  6,2,2,<CFE>,,DATSET,,<CFE>
     89
     90 000160  000005  000174' 000005  DATSET: .WORD   DEVL,DEVA,UICL,UICA,FILL,FILA
        000166  000201' 000012  000206'
     91
     92 000174     114     102     060  DEVA:   .ASCII  /LB00:/
        000177     060     072
     93         000005                  DEVL=   .-DEVA
     94
     95 000201     133     061     054  UICA:   .ASCII  /[1,2]/
        000204     062     135
     96         000005                  UICL=   .-UICA
     97
     98 000206     116     105     124  FILA:   .ASCII  /NETHLP.ULB/
        000211     110     114     120
        000214     056     125     114
        000217     102
     99         000012                  FILL=   .-FILA
    100
    101                                         .EVEN
```

GLOBAL DATA

```
    237             001574'                 $CT      == $COPES        ; COUNTER TIMER
    238             001570'                 $CHN     == $DPR          ; CHANNEL NUMBER
    239             001620'                 $HTM     == $VLVL         ; HELLO TIMER
    240             001564'                 $MBL     == $POLNA        ; MAXIMUM BLOCK
    241             001560'                 $MRC     == $SCSR         ; MAXIMUM RECALLS
    242             001556'                 $MWI     == $NN           ; MAXIMUM WINDOW
    243 001722                      $RTM::  .BLKW    1                ; RECALL TIMER/ROUTING TIMER
    244             001602'                 $HBT     == $TNUM         ; HOLDBACK TIMER
    245             001616'                 $RTC     == $EVBEG        ; MAXIMUM RETRANSMITS
    246             001556'                 $WSZ     == $NN           ; MAXIMUM WINDOW
    247             001564'                 $BSZ     == $POLNA        ; MAXIMUM BLOCK
    248             001576'                 $CUG     == $TNAM         ; CLOSED USER GROUP
    249             001576'                 $CALT    == $TNAM         ; CALL TIMER
    250             001562'                 $CLT     == $LINKS        ; CLEAR TIMER
    251             001602'                 $DFD     == $TNUM         ; DEFAULT DATA
    252             001540'                 $DFW     == $STAN         ; DEFAULT WINDOW
    253             001560'                 $MXC     == $SCSR         ; MAXIMUM CLEARS
    254             001550'                 $MRE     == $VEC          ; MAXIMUM RESETS
    255             001620'                 $RET     == $VLVL         ; RESET TIMER
    256             001570'                 $MXR     == $DPR          ; MAXIMUM RESTARTS
    257             001542'                 $MCI     == $NODNA        ; MAXIMUM CIRCUITS
    258             001550'                 $NUM     == $VEC          ; GROUP NUMBER
    259 001724                      $NNAM::                           ; NETWORK NAME
    260 001724                              .BLKW    1                ; LENGTH WORD
    261 001726                      $CLM::  .BLKW    11.              ; CALL MASK
    262 001754                              .BLKW    1                ; LENGTH WORD
    263 001756                      $CLV::  .BLKW    11.              ; CALL VALUE
    264 002004                      $SUB1:: .BLKW    1                ; LOW SUBADDRESS
    265 002006                      $SUB2:: .BLKW    1                ; HIGH SUBADDRESS
    266 002010                      $EAMAX::.BLKW    1                ; MAX AREA      (EXEC)
    267 002012                      $EACOS::.BLKW    1                ; AREA MAX COST (EXEC)
    268 002014                      $EAHOP::.BLKW    1                ; AREA MAX HOPS (EXEC)
    269 002016                      $EBRT:: .BLKW    1                ; BROADCAST ROUTING TIMER
    270 002020                      $EBEA:: .BLKW    1                ; BROADCAST ENDNODE ADJACENCIES (EXEC)
    271 002022                      $ESEG:: .BLKW    1                ; ECL SEGMENT SIZE (EXEC)
    272 002024                      $EINA:: .BLKB    1                ; INACTIVITY TIMER
    273 002025                      $EINC:: .BLKB    1                ; INCOMING TIMER
    274 002026                      $EOUT:: .BLKB    1                ; OUTGOING TIMER
    275 002027                      $ERET:: .BLKB    1                ; RETRANSMIT FACTOR
    276
    277                             ;
    278                             ; SNA-PROTOCOL OPTIONS -- OVERLAYING PSI CALL MASK AND CALL VALUE AREA
    279                             ;
    280             001724'         $SCIRC == $CLM-2                  ; CIRCUIT NAME (8 BYTES)
    281             001734'         $SACNM == $SCIRC+10               ; ACCESS NAME (8 BYTES)
    282             001744'         $SPLU  == $SACNM+10               ; PLU NAME (8 BYTES)
    283             001754'         $SMOD  == $SPLU+10                ; MODE TABLE (8 BYTES)
    284             001764'         $SUID  == $SMOD+10                ; USER ID (8 BYTES)
    285             001774'         $SPSW  == $SUID+10                ; PASSWORD (8 BYTES)
    286                             ;;;    == $SPSW+10                ; 4 SPARE BYTES
    287                             ;
    288                             ; SNA-PROTOCOL OPTIONS -- OVERLAYING EXECUTOR ID STRING
    289                             ;
    290             001622'         $SLUCT == $ID                     ; SLU RANGE COUNT (1 WORD)
    291             001624'         $SLUS  == $SLUCT+2                ; SLU LIST (16 WORDS)
    292                             ;
    293                             ; SNA-PROTOCOL OPTIONS -- OVERLAYING $VEC TO $TNUM INCLUSIVE
```

```
                                                                   I  1
NTIUMR     CREATED BY  MACRO  ON 29-JUN-85 AT 03:28       PAGE 1
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE              REFERENCES

M.BFVH    000011            6-87
M.BFVL    000012            6-86
M.UMRA    000002            6-88          6-90
M.UMRN    000004            6-89
$NIUMR    000000 RG        #6-84
```

```
                                                                   J  1
NTIUMR     CREATED BY  MACRO  ON 29-JUN-85 AT 03:28       PAGE 2
```

```
NTIX25     CREATED BY  MACRO  ON 29-JUN-85 AT 03:29     PAGE 3

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

ALUN$            #5-57      6-89
CALL              7-147     7-148      7-162      7-163      7-171      7-184
CALLR             7-185
DCBDF$           #5-57      5-63
DIR$             #5-56      7-125      7-156
EMSG$            #5-56
EMSG$R           #5-56      7-187      7-188      7-189
HDRDF$           #5-57      5-65
NHWDF$           #5-56      5-59
NSFDF$           #5-57      5-61
NTLER$           #5-56      6-95       6-96       6-97
OFF$             #6-88     #6-89
PDVDF$           #5-57      5-60
RESMAP           #5-56
RETC             #5-56      7-165
RETURN            7-151     7-168      7-175
RUN$             #5-56      6-88
R50$             #6-88      6-88
SAVMAP           #5-56
SWSTK$            7-147     7-162      7-184
UCBDF$           #5-57      5-64
VCBDF$           #5-57      5-62
```

```
NTICEX11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE                REFERENCES

BUFUMP   = 172354             #6-74
CMODE    = 140000             #6-74
FC.PCT   = 000030              6-94
FNDPDV   = ******   GX         8-188
FS.INI   = 000000              6-94
INIFUN     000000 R           #6-94          8-205
I$$AS    = ******              6-74          6-75
KISAR0   = 172340             #6-74
KISAR6   = 172354             #6-74
KSAR6    = ******   GX         8-228
LF.ACT   = 100000             #6-73
LF.BRO   = 000400             #6-73
LF.BWT   = 000007             #6-73
LF.ENA   = 002000             #6-73
LF.LPB   = 001000             #6-73
LF.MDC   = 000100             #6-73
LF.MFL   = 004000             #6-73
LF.MTP   = 000020             #6-73
LF.PAC   = 000200             #6-73
LF.RDY   = 040000             #6-73
LF.REA   = 010000             #6-73
LF.SER   = 000040             #6-73
LF.TIM   = 000010             #6-73
LF.UNL   = 020000             #6-73
LF.X2P   = 000000             #6-73
LLCAL    = ******   GX         8-207
LN.CLO   = 000000             #6-73
LN.DUM   = 000005             #6-73
LN.LOA   = 000004             #6-73
LN.LOO   = 000003             #6-73
LN.OAU   = 000003             #6-73
LN.OFF   = 000001             #6-73
LN.ON    = 000000             #6-73
LN.OOP   = 000004             #6-73
LN.OPE   = 000001             #6-73
LN.REF   = 000002             #6-73
LN.SER   = 000002             #6-73
LN.STA   = 000017             #6-73
LN.SUB   = 000360             #6-73
LN.TRI   = 000006             #6-73
L.COST     000015             #6-73
L.CTL      000012             #6-73
L.CVA      177776             #6-73
L.DDM      000002             #6-73
L.DDS      000004             #6-73
L.DLC      000003             #6-73
L.DLM      000006             #6-73
L.DLS      000010             #6-73
L.FLG      000000             #6-73
L.KRBA     000016             #6-73
L.LEN    = 000022             #6-73
L.MPF      000022             #6-73
```

```
NTICEX11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 2
```

```
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE             REFERENCES

ZF.LLC   = 000004          #6-63
ZF.LMC   = 000100          #6-63
ZF.MAN   = 020000          #6-63
ZF.MFL   = 000010          #6-63
ZF.MTM   = 000400          #6-63
ZF.MUX   = 000040          #6-63
ZF.PSE   = 002000          #6-63
ZF.SLI   = 010000          #6-63
ZF.TIM   = 000200          #6-63
ZF.X3P   = 000000          #6-63
ZS.ASN   = 100000          #6-63
ZS.BSY   = 140000          #6-63
Z.AVL      000014          #6-63
Z.DAT      000016          #6-63
Z.DSP      000000          #6-63      6-63
Z.FLG      000010          #6-63
Z.LEN    = 000016          #6-63
Z.LLN      000006          #6-63
Z.MAP      000020          #6-63
Z.NAM      000004          #6-63      7-165
Z.PCB      000012          #6-63
Z.SCH      000007          #6-63
$ERR20     000020 RG       #6-88      7-189
$NERDN   = ******   GX      6-88
$NICSR     000000 RG       #7-117
$NIERA   = ******   GX     *7-185    *7-186    *7-188
$NIERR   = ******   GX      6-88
$$$GLB   = ******           6-74      6-75
.BASEB   = 140000          #6-61      7-121     7-127     7-129     7-130
```

```
                .TITLE  NTILON - NTINIT 'EXECUTOR ON' ROUTINE
                .IDENT  /V05.00/

        ;
        ; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
        ;
        ;
        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
        ; TRANSFERRED.
        ;
        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
        ; CORPORATION.
        ;
        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
        ;
        ; MODULE DESCRIPTION:
        ;
        ;       NTINIT - ''SET EXECUTOR STATE ON'' PROCESSING
        ;
        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
        ;
        ; IDENT HISTORY:
        ;
        ; 1.00  27-FEB-78
        ;       VERSION 2.0 RELEASE
        ;
        ; 2.00  14-DEC-79
        ;       DECNET-11M/S V3.0
        ;       DECNET-11M-PLUS V1.0
        ;
        ; 3.00  16-APR-82
        ;       DECNET-11M V3.1
        ;       DECNET-11M-PLUS V1.1
        ;
        ; 4.00  07-NOV-83
        ;       DECNET-11M V4.0
        ;       DECNET-11M-PLUS V2.0
        ;
        ; 5.00  22-JUL-85
        ;       DECnet-11M/S V4.2
        ;       DECnet-11M-Plus V3.0
        ;       DECnet-Micro/RSX V1.0
        ;
        ;

```

```
NTILON11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:31      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE              REFERENCES

ALOCB    = ******   GX       9-254
AREAMK   = 176000           #7-86
ASSIGN     000036 R         #7-105        9-150
A.LULU   = 000002            7-105
BUFUMP   = 172354           #6-66
CMODE    = 140000           #6-66
DECPT    = ******   GX       9-157        9-246
D$NBRA     000054            9-247        9-249
D$NLN      000030           *9-158
E$NBR      000014           #6-71
E$NBS      000020           #6-71
E$NCR      000034           #6-71
E$NCS      000036           #6-71
E$NIC      000044           #6-71
E$NLEN     000050           #6-71
E$NLLA     000012           #6-71
E$NLNK     000000           #6-71
E$NML      000040           #6-71
E$NMR      000024           #6-71
E$NMS      000030           #6-71
E$NNOD     000002           #6-71
E$NRT      000042           #6-71
E$NRTP     000005           #6-71
E$NSEG     000010           #6-71
E$NTIM     000046           #6-71
E$NUSE     000004           #6-71
E$STRT     000006           #6-71
FNDPDV   = ******   GX       9-141
HEADR    = ******   GX       9-164
H.LUN      000076            9-165
I$$AS    = ******            6-66         6-70
KISAR0   = 172340           #6-66
KISAR6   = 172354           #6-66
LONACP     000010 R         #7-104        9-265
LVL1     = 002000           #7-87
MPAR     = 172100           #6-66
MPCSR    = 177746           #6-66
M$$MGE   = 000000            6-66
NF$MOU   = 000040            9-272
NF.END   = 000200            9-175
NF.LV2   = 000100            9-181
N$FLG      000005            9-272
N$PRI      000076           *9-262
PDVTA    = ******   GX       9-155
PIRQ     = 177772           #6-66
PMODE    = 030000           #6-66
PR0      = 000000           #6-66
PR1      = 000040           #6-66
PR2      = 000100           #6-66
PR3      = 000140           #6-66
PR4      = 000200           #6-66
PR5      = 000240           #6-66
```

```
NTILON11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:31      PAGE 2
```

```
    367                                   BNE     40$             ;; BR IF NO, TRY NEXT ONE
    368                                   CMPB    S$STA(R2),STANUM;; IS THIS THE CORRECT ONE?
    369                                   BNE     40$             ;; BR IF NO, TRY NEXT ONE
    370                                   MOV     @KSAR6,R3       ;; GET APR BIAS OF SVC DESCRIPTOR BLOCK
    371                           45$:    RESMAP                  ;; RESTORE PREVIOUS MAPPING
    372                                   MOV     R3,T$SVC(R0)    ;; STORE APR BIAS OF SVC DESCRIPTOR BLOCK
    373                                   MOV     R2,T$SVC+2(R0) ;; STORE VIRTUAL ADDRESS OF SVC BLOCK
    374                           50$:    RESMAP                  ;; RESTORE PREVIOUS MAPPING
    375                           60$:    RETURN
    376
    377                                   .ENDC
    378
    379         000001                    .END
```

```
                                .SBTTL  LOCAL DATA
                        ;
                        ; LOCAL DATA
                        ;
000000                          .PSECT  DATA,D

                                .IF DF  R$$MPL
                                .IF NDF R$$PRO
                        TRNVEC: GIN$    GI.VEC,EXEVEC,VECLN
                        VECLHD: .WORD   0
                                .ENDC
                                .ENDC

000000                  $NIECL::.BLKW                           ;HOLDS PDV INDEX FOR ECL
000002                  $NIXPT::.BLKW                           ;HOLDS PDV INDEX FOR XPT
000004                  $DBECL::.BLKW                           ;HOLDS ADDRESS OF ECL'S DDB
000006                  $DBXPT::.BLKW                           ;HOLDS ADDRESS OF XPT'S DDB

000010                  $NIERA::.BLKW   3                       ;ERROR REPORT ROUTINE ARGUMENT WORDS

000016                  NTIRCX: RCVX$   ,$NIRTN                 ;DIRECTIVE TO RCV INTO "$NIRTN" OR EXIT
000026                  $NIRTN::.BLKW   2                       ;REQUESTORS TASK NAME (FOLLOW BY $NIRDB)
000032                  $NIRDB::.BLKW   15.                     ;RCV DATA BLOCK (PRECEED BY $NIRTN)

               000001           MRKTEF = 1                      ;EVENT FLAG FOR MARK TIME
000070                  $NIMRK: MRKT$   MRKTEF,2,2              ;MARK TIME TO WAIT FOR EVL
000102                  $NIEVC::RQST$   EVC...                  ;DIRECTIVE TO REQUEST EVENT COLLECTOR

000120                  $NIAST: .BLKW   1                       ;NOTIFICATION AST ADDRESS
000122                  $NIPRM: .BLKW   1                       ;NOTIFICATION AST PARAMETER TO RETURN

                                .IF NDF R$$11S
                        NTIRCV: RCVD$   ,$NIRTN                 ;DIRECTIVE TO RCV DATA INTO "$NIRTN"

                        $NIRNL::RQST$   NTL...                  ;DIRECTIVE TO REQUEST NTL
                        $NISNL::SDAT$   NTL...,$NTLFC           ;DIRECTIVE TO SEND "$NTLFC" TO NTL
                        $NTLFC::.BLKW   3                       ;NTL FUNCTION CODE AND ANSWER

                        NTISEN::SDAT$   XXXXXX,$NIFNC           ;DIRECTIVE TO SEND DATA FROM "$NIFNC"
                        $NIFNC::.BLKW   3                       ;FUNCTION REQUEST SAVE WORD
                        NTIBSY: SDAT$   XXXXXX,NTIBSC           ;DIRECTIVE TO SEND DATA FROM "NTIBSC"
                        NTIBSC: .WORD   IE.ACT                  ;FAILURE CODE, NTI ALREADY BUSY

                        $NIFLG::.WORD   0                       ;FLAGS WORD
                                                                ;LOW BYTE:  NON-ZERO = BUSY W/ FUNCTION
                                                                ;HIGH BYTE: FUNCTION-SPECIFIC FLAGS

                                .IFF    ;NDF R$$11S

                                .IF DF  S$$BAS

                        NTISEN: SDAT$   XXXXXX,$NIFNC           ;DIRECTIVE TO SEND DATA FROM "$NIFNC"
                        $NIFNC::.BLKW   3                       ;FUNCTION REQUEST SAVE WORD

                        ;
                        ; THIS TABLE MAPS THE NTINIT ERROR MESSAGES TO THE DECIMAL NUMBER TO
                        ; BE DISPLAYED ON THE LEDS FOR SERVER BASE SYSTEMS
                        ;
```

```
                                                .SBTTL  ERROR MESSAGE PROCESSOR

                                        ;+
                                        ; $NIERR - FORMAT AND PRINT AN ERROR MESSAGE
                                        ;
                                        ; THIS ROUTINE DOES NOT NOW SUPPORT SECOND LINE FORMATTING AND/OR PRINTING
                                        ;
                                        ; INPUTS:
                                        ;       $NIERA  = ARGUMENTS FOR ASTERISK REPLACER (IF ONE SPECIFIED)
                                        ;       R0      = ADDRESS OF SECOND WORD OF ERROR CONTROL BLOCK
                                        ;       BLOCK FORMAT:
                                        ;               .WORD   A       ;ADDRESS OF ERROR PROCESSOR ($NIERR)
                                        ;               .WORD   0       ;ADDRESS OF FORMAT STRING (NOT USED)
                                        ;               .WORD   B OR 0  ;ADDRESS OF ASTERISK REPLACEMENT ROUTINE
                                        ;               .WORD   0       ;ADDRESS OF SECOND LINE FORMATTER (NOT USED)
                                        ;               .ASCIZ  /MESS/  ;ERROR MESSAGE
                                        ;
                                        ; OUTPUTS:
                                        ;       MESSAGE IS PRINTED, EXCEPT FOR SERVER BASE WHEN EXEC $HALT ROUTINE USED
                                        ;       R0 IS DESTROYED
                                        ;       CARRY IS SET
                                        ;-

 000332 010146                          $NIERR::MOV     R1,-(SP)                ;SAVE R1
 000334 010246                                  MOV     R2,-(SP)                ;SAVE R2
 000336 010346                                  MOV     R3,-(SP)                ;SAVE R3
 000340 010446                                  MOV     R4,-(SP)                ;SAVE R4
 000342 010546                                  MOV     R5,-(SP)                ;SAVE R5

                                                .IF DF  S$$BAS

                                                TST     -(R0)                   ;POINT TO FIRST WORD IN FORMAT BLOCK
                                                MOV     #ERRTAB,R1              ;POINT TO ERROR TABLE
                                        10$:    MOV     (R1)+,R2                ;GET NEXT ENTRY IN TABLE (FORMAT ADDRESS)
                                                CMP     R2,#TABEND              ;END OF TABLE?
                                                BEQ     30$                     ;BR IF END OF TABLE
                                                CMP     R2,R0                   ;IS THIS THE CORRECT ENTRY?
                                                BEQ     20$                     ;BR IF YES
                                                TST     (R1)+                   ;SKIP LED ERROR CODE
                                                BR      10$                     ;CONTINUE
                                        ;
                                        ; FOUND CORRECT ENTRY
                                        ;
                                        20$:    MOV     (R1)+,R1                ;GET LED ERROR CODE

                                                SWSTK$  30$                     ;;ENTER SYSTEM STATE TO HALT PROCESSOR
                                                CALL    $HALT                   ;;JUMP TO ROUTINE TO HALT PROCESSOR
                                                                                ;;AND DISPLAY ERROR - NEVER RETURNS!!
                                        ;
                                        ; END OF TABLE
                                        ;
                                        30$:                                    ;END OF TABLE - IGNORE ERROR

                                                .IFF    ;DF S$$BAS

 000344 010005                                  MOV     R0,R5                   ;COPY ADDRESS IN ERROR BLOCK
 000346 012700  000137'                         MOV     #NTIERB,R0              ;GET ADDRESS OF MESSAGE BUFFER
```

```
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE            REFERENCES

SF.ENA  = 000100         #6-76
SF.LPB  = 000004         #6-76
SF.MFL  = 000040         #6-76
SF.PAC  = 000020         #6-76
SF.REA  = 000010         #6-76
SF.SER  = 000001         #6-76
SF.SVC  = 000002         #6-76
SF.UNL  = 000040         #6-76
SLTMA     000464 RG      #8-232
SLTNM     000466 RG      #8-233
SRSTD     000434 RG      #8-215
SWR     = 177570         #6-77
S$$BAS  = ******          7-132      8-192      8-192      8-193      8-193      8-194      8-194      9-312      10-373
                          10-407     10-452     14-642     15-711     18-848
S.COST    000001         #6-76
S.FLG     000000         #6-76
S.LEN     000004         #6-76
S.NMST    000002         #6-76
S.OWNR    000003         #6-76
TKPS      000454 RG      #8-223
TKTCB     000446 RG      #8-220      10-411
TK100     000500 RG      #8-238
TPS     = 177564         #6-77
T.NAM     000006         *10-413
T.ST2     000034         *10-412
T1SCL     000512 RG      #8-243
T100C     000474 RG      #8-236
T100Q     000476 RG      #8-237
T2.FXD  = 002000          10-412
UBMPR   = 170200         #6-77
UISAR0  = 177640         #6-77
UISAR1  = 177642         #6-77
UMRPT     000432 RG      #8-214
VECLN   = 000043   G     #8-252
X$$MCB  = ******          6-75       6-75
ZF.COU  = 001000         #6-75
ZF.DDM  = 000001         #6-75
ZF.DIA  = 004000         #6-75
ZF.DLC  = 000002         #6-75
ZF.DVP  = 100000         #6-75
ZF.INI  = 040000         #6-75
ZF.KMX  = 000020         #6-75
ZF.LLC  = 000004         #6-75
ZF.LMC  = 000100         #6-75
ZF.MAN  = 020000         #6-75
ZF.MFL  = 000010         #6-75
ZF.MTM  = 000400         #6-75
ZF.MUX  = 000040         #6-75
ZF.PSE  = 002000         #6-75
ZF.SLI  = 010000         #6-75
ZF.TIM  = 000200         #6-75
ZF.X3P  = 000000         #6-75
```

```
                                                                   I 11
NTITIM11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:33      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE                REFERENCES

ABTIM    = ******   GX     9-178
BUFUMP   = 172354         #6-62
CLINS      000162 R        8-152     #9-173
CLKHD    = ******   GX     7-94       9-182
CMODE    = 140000         #6-62
C.TIM      000006          9-186      9-189
I$$AS    = ******          6-62
KISAR0   = 172340         #6-62
KISAR6   = 172354         #6-62
MPAR     = 172100         #6-62
MPCSR    = 177746         #6-62
M$$MGE   = 000000          6-62
PIRQ     = 177772         #6-62
PMODE    = 030000         #6-62
PR0      = 000000         #6-62
PR1      = 000040         #6-62
PR2      = 000100         #6-62
PR3      = 000140         #6-62
PR4      = 000200         #6-62
PR5      = 000240         #6-62
PR6      = 000300         #6-62
PR7      = 000340         #6-62       7-98
PS       = 177776         #6-62      *7-98      *7-111
R$$MPL   = ******          6-62       6-62       6-62       10-198
R$$11D   = ******          6-62
R$$11S   = 000001         #4-2
SWR      = 177570         #6-62
TIFLG      000000 R       #6-72      *7-103     *7-108      7-112      7-115
TKPS     = ******   GX     8-143
TK100    = ******   GX     8-149
TPS      = 177564         #6-62
T1       = 000002         #6-70       7-108      7-115
T1SCL    = ******   GX     7-106      8-142
T1SIN      000116 R        7-117     #8-141
T100C    = ******   GX     7-101      8-148
T100Q    = ******   GX     8-146
T5       = 000001         #6-69       7-103      7-112
T50IN      000130 R        7-114     #8-145
UBMPR    = 170200         #6-62
UISAR0   = 177640         #6-62
UISAR1   = 177642         #6-62
$NITIM     000002 RG      #7-94
.BASEB   = 140000         #6-62
```

```
                                                                   J 11
NTITIM11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:33      PAGE 2
```

Table of contents

```
     5-  109    $CFCOM - GET A COMMAND LINE
     6-  172    $CLERR - CLEAR ERROR MESSAGE ADDRESS
```

```
    110                                          ;
    111                                          ; ERROR TEXT
    112                                          ;
    113             000002                              EX$ERR  = 2
    114                                                 .NLIST  BEX
    115 000034         002       103       106  ERRIWC: .ASCIZ  <EX$ERR>/CFE -- Parameter illegal for wildcard operation/
    116 000115         002       103       106  ERRIGR: .ASCIZ  <EX$ERR>/CFE -- Parameter illegal for circuit type/
    117 000170         002       103       106  ERRPSI: .ASCIZ  <EX$ERR>/CFE -- Parameter applicable to PSI systems only/
    118                                                 .EVEN
    119                                                 .LIST   BEX
    120
```

```
                                                                 I 14
CFEDEF     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 2

SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE            REFERENCES

SW.SLT  = ******  GX      9-345
WC.CIR  = ******  GX      8-276
XP.CUG  = ******  GX      12-440
XP.DTE  = ******  GX      12-437
XP.KND  = ******  GX      12-437
XP.KNG  = ******  GX      12-440
XP.LV3  = ******  GX      12-443
XP.NNA  = ******  GX      12-433
$CFDEF    000252 RG      #5-134
$CFERR  = ******  GX      6-230     8-321     8-324     8-327     12-448
$COMP   = ******  GX      5-135
$ESUB   = ******  GX      6-186     6-187
$FLAGS  = ******  GX      8-298     8-302    *8-304     8-307
$NNAM   = ******  GX      12-435    12-436
$OPRND  = ******  GX      6-152     6-159     6-165     6-169     7-246     7-249     7-253     7-256     8-282
                         *8-290     8-315     9-347     9-350     9-354     10-385   *10-386    10-387    10-390
                         *10-393    11-412    11-415    12-433    12-437    12-440    12-443
$OPRN2  = ******  GX      6-154     6-161
$PSIPT  = ******  GX      6-182     8-300     12-431
$SEBUF  = ******  GX      11-417
$SECNT  = ******  GX      7-251     9-352
$SECUG  = ******  GX      12-442
$SEDTE  = ******  GX      12-439
$SELOG  = ******  GX      4-101
$SENOD  = ******  GX      6-167
$SENSP  = ******  GX      6-163
$SEOBJ  = ******  GX      4-99
$SEPAR  = ******  GX      11-414
$SEPRO  = ******  GX      4-96
$SEPVC  = ******  GX      10-375
$SEREM  = ******  GX      4-95
$SEROU  = ******  GX      6-156
$SESLT  = ******  GX      7-248     9-349     10-392
$SESNA  = ******  GX      4-108
$SESVC  = ******  GX      10-377
$SEUNT  = ******  GX      7-255     9-356     10-389
$SEXAC  = ******  GX      4-106
$SEXSV  = ******  GX      4-105
$SEX2P  = ******  GX      7-258
$SEX29  = ******  GX      4-107
$SEX3P  = ******  GX      12-445
$SWICH  = ******  GX      9-345
$WILD   = ******  GX      8-276
```

```
                                                                 J 14
CFEDEF     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 3
```

```
                                                .SBTTL  $CFHLP - PROCESS HELP COMMAND
                                        ;+
                                        ; *** - $CFHLP - PROCESS HELP COMMAND
                                        ;
                                        ; INPUTS:
                                        ;       NONE
                                        ;
                                        ; OUTPUTS:
                                        ;       HELP TEXT PRINTED
                                        ;
                                        ;-

000220                                  $CFHLP::
000220  016704  000000G                         MOV     $CMDPT,R4               ; POINT AT LINE AFTER "HELP"
000224                                          CALLR   $HELP                   ; PRINT HELP INFO


        000001                                  .END
```

```
                                 ;
          001550'                $SDAT  == $VEC               ; USER DATA STRING (33 BYTES ASCIZ)
                                 ;
                                 ; SNA PU OPTIONS -- OVERLAYING PSI CALL MASK AND VALUE AREA
                                 ;
          001724'                $SPUNM == $CLM-2             ; CIRCUIT NAME (8 BYTES) - SAME AS $SCIRC
          001734'                $SPARE == $SPUNM+10          ; SPARE - FOR FMTLIN WORK SPACE (8 BYTES)
          001744'                $SNLU  == $SPARE+10          ; MAXIMUM LU COUNT (1 WORD)
          001746'                $SNSID == $SNLU+2            ; SSCP ID COUNT (1 WORD)
          001750'                $SSERN == $SNSID+2           ; SDLC SERIAL NUMBER (2 WORDS)
          001754'                $SSSCP == $SSERN+4           ; SSCP IDENT LIST (8 WORDS)
                                 ;;;    == $SSSCP+20          ; 4 SPARE WORDS
                                 ;
                                 ; REMOTE NODE OPTIONS - OVERLAYS EVERYTHING FOLLOWING $VEC
                                 ;
          001552'                $RNHOS  == $VEC+2            ;Host node address
          001554'                $RNSDV  == $RNHOS+2          ;Service device
          001556'                $RNDPA  == $RNSDV+2          ;Dump address
          001562'                $RNDCP  == $RNDPA+4          ;Dump count
          001564'                $RNSCI  == $RNDCP+2          ;Service circuit
          001574'                $RNHAR  == $RNSCI+10         ;Hardware address
          001604'                $RNSPA  == $RNHAR+10         ;Service password
          001620'                $RNSLO  == $RNSPA+14         ;Secondary loader file
          001657'                $RNTLO  == $RNSLO+31.        ;Tertiary loader file
          001716'                $RNLOD  == $RNTLO+31.        ;Load file
          001755'                $RNDIA  == $RNLOD+31.        ;Diagnostics file
          002014'                $RNDMP  == $RNDIA+31.        ;Dump file
          002053'                $RNXXX  == $RNDMP+31.        ;-end of remote node options-

  002030                         .IIF LT <.-$RNXXX>  .BLKB <$RNXXX-.>

                                         .EVEN                ; PHEW !
```

J 1

```
NTIUMR      CREATED BY  MACRO  ON 29-JUN-85 AT 03:28      PAGE 2
MACRO CROSS REFERENCE                                     CREF   04.00
MACRO NAME        REFERENCES

RETURN             6-100
SCBDF$            #5-59         5-61
```

K 1

```
**FILE**ID**NTIX25
```

```
NNN         NNN     TTTTTTTTTTTTTTT        IIIIIIIII              111                 111              SSSSSSSSSSSS
NNN         NNN     TTTTTTTTTTTTTTT        IIIIIIIII              111                 111              SSSSSSSSSSSS
NNN         NNN     TTTTTTTTTTTTTTT        IIIIIIIII              111                 111              SSSSSSSSSSSS
NNN         NNN           TTT                 III              111111              111111           SSS
NNN         NNN           TTT                 III              111111              111111           SSS
NNN         NNN           TTT                 III              111111              111111           SSS
NNNNNN      NNN           TTT                 III                 111                 111           SSS
NNNNNN      NNN           TTT                 III                 111                 111           SSS
NNNNNN      NNN           TTT                 III                 111                 111           SSS
NNN   NNN   NNN           TTT                 III                 111                 111              SSSSSSSSS
NNN   NNN   NNN           TTT                 III                 111                 111              SSSSSSSSS
NNN   NNN   NNN           TTT                 III                 111                 111              SSSSSSSSS
NNN      NNNNNN           TTT                 III                 111                 111                       SSS
NNN      NNNNNN           TTT                 III                 111                 111                       SSS
NNN      NNNNNN           TTT                 III                 111                 111                       SSS
NNN         NNN           TTT                 III                 111                 111                       SSS
NNN         NNN           TTT                 III                 111                 111                       SSS
NNN         NNN           TTT                 III                 111                 111                       SSS
NNN         NNN           TTT              IIIIIIIII           111111111           111111111        SSSSSSSSSSSS
NNN         NNN           TTT              IIIIIIIII           111111111           111111111        SSSSSSSSSSSS
NNN         NNN           TTT              IIIIIIIII           111111111           111111111        SSSSSSSSSSSS
```

K 2

FILEID**NTICEX

```
                                                                     J  3
NTICEX11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF    04.00
SYMBOL   VALUE              REFERENCES

L.NMST     000020           #6-73
L.NSTA     000014           #6-73
L.OWNR     000021           #6-73
L.UNT      000013           #6-73
MPAR     = 172100           #6-74
MPCSR    = 177746           #6-74
M$$EXT   = 000000  G        #6-78       8-230       8-237
M$$MGE   = 000000            6-74
NTILIN     000000 RG        #8-188
NTLPT    = ******  GX        8-228
PDVID    = ******  GX        8-204
PDVNM    = ******  GX        8-191
PDVTA    = ******  GX        8-190
PIRQ     = 177772           #6-74
PMODE    = 030000           #6-74
PR0      = 000000           #6-74
PR1      = 000040           #6-74
PR2      = 000100           #6-74
PR3      = 000140           #6-74
PR4      = 000200           #6-74
PR5      = 000240           #6-74
PR6      = 000300           #6-74
PR7      = 000340           #6-74
PS       = 177776           #6-74
PUMR     = ******  GX        8-260
R$$MPL   = ******            6-67       6-74        6-74        6-74        6-96        8-222       8-242       8-254       8-273
                             9-300
R$$11D   = ******            6-74       6-75
R$$11M   = 000000            6-75
R$$11S   = 000001           #4-2        6-75        6-88        7-135       8-216
SF.ACT   = 000200           #6-73
SF.ENA   = 000100           #6-73
SF.LPB   = 000004           #6-73
SF.MFL   = 000040           #6-73
SF.PAC   = 000020           #6-73
SF.REA   = 000010           #6-73
SF.SER   = 000001           #6-73
SF.SVC   = 000002           #6-73
SF.UNL   = 000040           #6-73
SWR      = 177570           #6-74
S.COST     000001           #6-73
S.FLG      000000           #6-73
S.LEN      000004           #6-73
S.NMST     000002           #6-73
S.OWNR     000003           #6-73
TPS      = 177564           #6-74
UBMPR    = 170200           #6-74
UISAR0   = 177640           #6-74
UISAR1   = 177642           #6-74
UMRPT    = ******  GX        8-246
X$$MCB   = ******            6-75       6-75
ZF.COU   = 001000           #6-75
```

```
                                                                     K  3
NTICEX11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 3
```

```
NTICSR11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 4
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              7-123       7-189
DIR$             #6-59        7-149       7-151
EMSG$            #6-59        7-189
NHWDF$           #6-59        6-61
NTLER$           #6-59        6-88
OFF$             #6-74       #6-75
PDVDF$           #6-59        6-63
RESRG            #6-59        7-190       7-198
RETURN            7-143       7-199
SAVRG            #6-59        7-119       7-161
SLTDF$           #6-59        6-62
SVTK$            #6-59        6-74        6-75
SWSTK$            7-123
```

```
**FILE**ID**NTIDLI
```

```
     54                                         .SBTTL  MACROS
     55
     56
     57                                 ;
     58                                 ; LIBRARY MACROS
     59                                 ;
     60                                         .MCALL  DIR$,RUN$,EMSG$,EMSG$R,NTLER$,NHWDF$,RETC,WSIG$S
     61                                         .MCALL  ALUN$,PLBDF$,XPDDB$,ECDDB$,PDVDF$,CTRDF$,MAP,BIAS
     62                                         .MCALL  RESMAP,SAVMAP,ADJDF$,NSFDF$,CALLX,CACDF$,DHBDF$
     63                                         .MCALL  DVBDF$,DCBDF$,UCBDF$,HDRDF$,CCBDF$
     64
     65 000000                                  CCBDF$                          ;DEFINE FUNCTION CODE SYMBOLS
     66 000000                                  NHWDF$                          ;HARDWARE REGISTER DEFINITIONS
     67 000000                                  PLBDF$                          ;DEFINE PLB OFFSETS
     68 000000                                  XPDDB$                          ;DEFINE TRANSPORT DATABASE OFFSETS
     69 000000                                  ECDDB$                          ;DEFINE ECL DATABASE OFFSETS
     70 000000                                  PDVDF$                          ;DEFINE PDV TABLE OFFSETS
     71 000000                                  CTRDF$                          ;DEFINE NSP COUNTER OFFSETS
     72 000000                                  NSFDF$                          ;DEFINE NETWORK FLAGS
     73 000000                                  ADJDF$                          ;DEFINE ADJACENCY DATA BASE OFFSETS
     74 000000                                  CACDF$                          ;DEFINE NI CACHE DATA BASE OFFSETS
     75 000000                                  DHBDF$                          ;DEFINE DECNET HOME BLOCK OFFSETS
     76 000000                                  DVBDF$                          ;VCB DEFINITIONS
     77 000000                                  DCBDF$                          ;DCB DEFINITIONS
     78 000000                                  UCBDF$                          ;UCB DEFINITIONS
     79 000000                                  HDRDF$                          ;TASK HEADER OFFSETS
```

```
                                                             J  6
NTILON11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:31     PAGE 2

SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL   VALUE                REFERENCES

PR6     = 000300             #6-66
PR7     = 000340             #6-66
PS      = 177776             #6-66
RCP1NM    000000 R           #7-101     9-177
RCP2NM    000004 R           #7-102     9-184
R$$MPL  = ******              6-66      6-66      6-66      9-163
R$$11D  = ******              6-66      6-70
R$$11M  = 000000              6-70
R$$11S  = 000001             #4-2       6-70      7-96      9-198     9-283     10-313
R.UNTN  = 000002              7-104
SRSTD   = ******   GX         9-178     9-185
SWR     = 177570             #6-66
S$$BAS  = ******              8-112     8-112     8-113     8-113     8-114     8-114     8-115     8-115     8-116
                              8-116
TPS     = 177564             #6-66
T$FLAG    000044             #6-71
T$LIF     000013             #6-71
T$LIFL    000013             #6-71
T$LIFO    000013             #6-71
T$LIFS    000013             #6-71
T$LIN     000000             #6-71
T$LIPS    000006             #6-71
T$LLD     000012             #6-71
T$LLDC    000045             #6-71
T$LLDL    000012             #6-71
T$LLDO    000012             #6-71
T$LLDS    000012             #6-71
T$LLEN    000046             #6-71
T$LOPR    000002             #6-71
T$LTCL    000024             #6-71
T$LTIM    000026             #6-71
T$LTPR    000014             #6-71
T$LTPS    000020             #6-71
T$NAPL    000004             #6-71
T$NFE     000000             #6-71
T$NLEN    000010             #6-71
T$NNUL    000002             #6-71
T$NOPL    000006             #6-71
T$NRNI    000042             #6-71
T$NRPL    000005             #6-71
T$NRUL    000007             #6-71
T$NVR     000001             #6-71
T$RPRI    000040             #6-71
T$SVC     000034             #6-71
T$T5      000030             #6-71
T$T6      000032             #6-71
UBMPR   = 170200             #6-66
UISAR0  = 177640             #6-66
UISAR1  = 177642             #6-66
U.CW3     000014              9-174
X$$MCB  = ******              6-70      6-70
ZF.COU  = 001000             #6-70
```

```
                                                             K  6
NTILON11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:31     PAGE 3
```

```
                                                      J  7
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:52  Page 9-2
Symbol table

ACTCIR  000004R         CP.DCF= 000040          C.URM   177776          E$NIC   000044          FS.STP= 002000
AT$ACL= 000100          CP.HDL= 000007          C.XACP  000004          E$NLEN  000050          FS.STR= 001000
AT$ADN= 000040          CP.PS = 177400          C.XID   000035          E$NLLA  000012          FS.TRM= 003000
AT$AUP= 000020          CP.PSI= 000200          C.XLEN  000044          E$NLNK  000000          FS.WLB= 001000
AT$CYC= 000004          CP.XCF= 000100          C.XPLI  000040          E$NML   000040          FS.XKL= 002000
AT$LV1= 000002          CP.2FR= 000030          C.XPT   000034          E$NMR   000024          FS.XOF= 010000
AT$LV2= 000001          CS.ABO= 000100          C.XSVC  000042          E$NMS   000030          FS.XON= 007000
AT$NEX= 000200          CS.BRO= 000002          C.XTC   000037          E$NNOD  000002          FS.ZER= 002000
AT$UP = 000010          CS.BUF= 000200          C.X25   000036          E$NRT   000042          F$$LVL= 000001
A$CIR   000003          CS.CES= 000002          DECPT = ****** GX       E$NRTP  000005          G$$TPP= 000000
A$LEN   000004          CS.CHN= 000010          D$AMXC  000072          E$NSEG  000010          G$$TSS= 000000
A$NID   000000          CS.CMP= 000200          D$AMXH  000074          E$NTIM  000046          G$$TTK= 000000
A$TM    000003          CS.DCR= 000400          D$ANN   000000          E$NUSE  000004          G$$WRD= 000000
A$TMI   000002          CS.DEF= 000004          D$BRPR  000102          E$STRT  000006          HF$DLM= 000002
A$TSZ   000000          CS.DEV= 000002          D$BRTM  000100          E$$XPR= 000000          HF$GWY= 000010
A$TYP   000002          CS.DIS= 000040          D$DELF  000045          FC.CCP= 000020          HF$HOS= 000004
A$$CHK= 000000          CS.ENA= 000001          D$DELW  000046          FC.CTL= 000006          HF$XDF= 000020
A$$CPS= 000000          CS.ENB= 000020          D$END = 000104          FC.KCP= 000016          H$CUG   000010
A$$PRI= 000000          CS.ERR= 100000          D$FNB   000034          FC.KIL= 000004          H$DST   000012
A$$TRP= 000000          CS.FTL= 001000          D$HIOR  000024          FC.MAN= 000024          H$D29   000014
CB.CCB= 000002          CS.HCR= 000001          D$HOST  000022          FC.MLD= 000026          H$FLG   000000
CB.DDM= 000040          CS.HFE= 002000          D$INAC  000044          FC.PCT= 000030          H$GLEN  000104
CB.DLC= 000020          CS.LST= 040000          D$INCT  000042          FC.PWR= 000022          H$GLT   000044
CB.RDB= 000004          CS.MTL= 004000          D$IPL   000051          FC.RCE= 000002          H$GNAM  000050
CB.SDB= 000010          CS.RNG= 000010          D$LID   000020          FC.RCP= 000014          H$GNML= 000020
CB.SLI= 000100          CS.ROV= 000004          D$LNAM  000006          FC.TIM= 000010          H$GPT   000046
CB.XLB= 000001          CS.RSN= 010000          D$LNUM  000014          FC.XCP= 000012          H$HITS  000034
CCBGT = ****** GX       CS.SHU= 000001          D$LST   000047          FC.XME= 000000          H$HLEN  000044
CC.LLC= 000200          CS.SID= 000002          D$MAXC  000064          FNDPDV= ****** GX       H$LBDA  000070
CE.ABO= 100362          CS.STR= 000004          D$MAXH  000066          FS.AST= 000000          H$LBDN  000072
CE.DAO= 100346          CS.SUC= 000001          D$MAXV  000070          FS.CIB= 002000          H$LDTE  000002
CE.DIS= 100366          CS.TMO= 020000          D$MLL   000040          FS.CRA= 001000          H$LEN   000042
CE.ERR= 100370          CS.XUR= 000004          D$MNOD  000041          FS.DIS= 013000          H$LOTS  000032
CE.ILN= 100350          CTLNUM  000003R         D$NA    000062          FS.DVC= 001000          H$NETW  000024
CE.LTO= 100356          C$$CKP= 000000          D$NBEA  000056          FS.ENB= 012000          H$NML = 000006
CE.MOP= 100372          C$$ORE= 000400          D$NBRA  000054          FS.EXI= 001000          H$NPT   000022
CE.NTE= 100361          C$$RSH= 177564          D$NEND= 000054          FS.GET= 006000          H$PTB   000020
CE.RTE= 100376          C.ADD   000034          D$NLN   000030          FS.HLT= 000000          H$PVC   000006
CE.SRC= 100364          C.BID   000003          D$NN    000060          FS.INI= 000000          H$RDTE  000004
CE.STP= 100352          C.BUF   000014          D$OUTT  000043          FS.KIL= 000000          H$RNW   000042
CE.TME= 100354          C.BUF1  000014          D$RETF  000050          FS.LCL= 100000          H$SVC   000036
CE.TMO= 100374          C.BUF2  000024          D$RNN   000002          FS.LTM= 001000          H$TRB   000016
CE.UNS= 100344          C.CNT   000020          D$RTMR  000076          FS.MNT= 004000          H$XAVL  000100
CF.CHN= 000001          C.CNT1  000020          D$SEG   000036          FS.MSN= 014000          H$XBIA  000074
CF.EOM= 000004          C.CNT2  000030          D$SER   000032          FS.REA= 001000          H$X29C  000040
CF.HDR= 000020          C.FLG   000022          D$SQRL  000052          FS.RET= 000000          I$$RAR= 000000
CF.LB = 100000          C.FLG1  000022          D$$BUG= 177514          FS.REZ= 003000          I$$RDN= 000000
CF.LIN= 000002          C.FLG2  000032          D$$ISK= 000000          FS.RLB= 002000          K$$CNT= 177546
CF.SOM= 000010          C.FNC   000010          D$$L11= 000001          FS.RNG= 011000          K$$CSR= 177546
CF.SYN= 000040          C.LIN   000006          D$$YNC= 000000          FS.RST= 000000          K$$LDC= 000000
CF.TRN= 000100          C.LNK   000000          D$$YNM= 000000          FS.RTN= 001000          K$$TPS= 000074
CM.CIR= 000002          C.MOD   000011          ENAC    000074R         FS.SET= 005000          LD$LP = 000000
CM.FMT= 100000          C.NSP   000004          ENALIN  000610R         FS.SFC= 005000          LF.ACT= 100000
CM.HRD= 000002          C.PRO   000042          E$NBR   000014          FS.SFR= 006000          LF.BRO= 000400
CM.LIN= 000000          C.RSV   000002          E$NBS   000020          FS.SFS= 004000          LF.BWT= 000007
CM.LOO= 000001          C.STA   000007          E$NCR   000034          FS.SPW= 040000          LF.ENA= 002000
CM.XLO= 000004          C.STS   000012          E$NCS   000036          FS.STM= 000000          LF.LPB= 001000
```

```
                                                      K  7
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:52  Page 9-3
```

```
                        ERRTAB: .WORD   $ERR04,20.              ;COMM EXEC NOT LOADED
                                .WORD   $ERR11,21.              ;ROUTER PRI DATA BASE ALLOCATION FAILURE
                                .WORD   $ERR15,24.              ;NS: ASSIGNMENT FAILURE
                                .WORD   $ERR17,25.              ;NETACP NOT INSTALLED
                                .WORD   $ERR19,26.              ;RCP NOT INSTALLED
                                .WORD   $ERR20,27.              ;DEVICE OFFLINE
                                .WORD   TABEND                  ;END OF TABLE

                                TABEND = -1                     ;END OF TABLE CONSTANT DEFINITION

                                .IFF

        000032'         $NIFNC == $NIRDB

                                .ENDC   ;DF S$$BAS

                                .ENDC   ;NDF R$$11S
```

```
 000352   005725                       TST     (R5)+                   ;SKIP ADDRESS OF FORMAT STRING
 000354   010504                       MOV     R5,R4                   ;COPY ADDRESS AGAIN
 000356   022424                       CMP     (R4)+,(R4)+             ;POINT AT ERROR MESSAGE

 000360   112410             10$:      MOVB    (R4)+,(R0)              ;COPY A BYTE INTO BUFFER
 000362   001412                       BEQ     20$                     ;BR IF END OF MESSAGE
 000364   122720   000052              CMPB    #'*,(R0)+               ;FOUND AN ASTERISK?
 000370   001373                       BNE     10$                     ;BR IF NO
 000372   005715                       TST     (R5)                    ;IS THERE AN ASTERISK REPLACER?
 000374   001402                       BEQ     15$                     ;BR IF NO
 000376   005300                       DEC     R0                      ;OVERWRITE THE ASTERISK
 000400                                CALL    @(R5)+                  ;CALL THE REPLACER

 000402   112420             15$:      MOVB    (R4)+,(R0)+             ;COPY REST OF MESSAGE
 000404   001376                       BNE     15$                     ;UNTIL A NULL BYTE
 000406   005300                       DEC     R0                      ;BACK UP TO NULL

 000410   010005             20$:      MOV     R0,R5                   ;COPY ADDRESS OF END OF BUFFER
 000412   162705   000124'             SUB     #NTIERN,R5              ;GET LENGTH OF MESSAGE
 000416                                QIOW$S  #IO.WVB,#1,#2,,,,<#NTIERN,R5,#40>

 000466   012605                       MOV     (SP)+,R5                ;RECOVER R5
 000470   012604                       MOV     (SP)+,R4                ;RECOVER R4
 000472   012603                       MOV     (SP)+,R3                ;RECOVER R3
 000474   012602                       MOV     (SP)+,R2                ;RECOVER R2
 000476   012601                       MOV     (SP)+,R1                ;RECOVER R1
 000500   000261                       SEC
 000502                                RETURN

                                       .ENDC   ;DF S$$BAS


```

```
NTINIT11S  CREATED BY  MACRO  ON 5-SEP-85 AT 11:10      PAGE 5
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE                REFERENCES

ZS.ASN  = 100000              #6-75
ZS.BSY  = 140000              #6-75
Z.AVL     000014              #6-75
Z.DAT     000016              #6-75        13-599       13-609
Z.DSP     000000              #6-75        6-75
Z.FLG     000010              #6-75
Z.LEN   = 000016              #6-75
Z.LLN     000006              #6-75
Z.MAP     000020              #6-75
Z.NAM     000004              #6-75
Z.PCB     000012              #6-75        13-597       13-607
Z.SCH     000007              #6-75
$ABTIM  = ******    GX          8-224
$ALOCB  = ******    GX          8-216
$CALLX  = ******    GX          8-235
$CBDMG  = ******    GX          15-730     15-742
$CCBGT  = ******    GX          8-239
$CEACC  = ******    GX          8-242
$CLKHD  = ******    GX          8-222
$C5TA   = ******    GX          15-716
$DBECL    000004 RG           #7-99       *13-587      *13-599
$DBXPT    000006 RG           #7-100      *13-588      *13-609
$DEACB  = ******    GX          8-217
$DECPT  = ******    GX          8-245
$DEVHD  = ******    GX          8-225
$ERR03    000246 R            #8-192       10-378
$ERR04    000300 R            #8-193       10-380
$ERR12    000336 RG           #8-194
$FMASK  = ******    GX          8-203
$HEADR  = ******    GX          8-219
$LDPWF  = ******    GX          8-226
$LGSTT  = ******    GX          8-244
$LLCAL  = ******    GX          8-248
$LLCRS  = ******    GX          8-240
$LLCTA  = ******    GX          8-249
$NERDD    000550 RG           #15-737
$NERDN    000526 RG           #15-726      15-737
$NER50    000504 RG           #15-715      15-726
$NIAST    000120 R            #7-112
$NICEX  = ******    GX          10-387
$NIECL    000000 RG           #7-97       *13-585      *13-594
$NIERA    000010 RG           #7-102       15-715       15-728       15-738
$NIERR    000332 RG             8-192      8-193        8-194       #14-636
$NIEVC    000102 RG           #7-110       10-403
$NIFNC  = 000032 RG           #7-153       10-361       10-389       10-409      *10-434      *10-436
$NILON  = ******    GX          10-404
$NIMRK    000070 R            #7-109       10-401
$NIPRM    000122 R            #7-113
$NIRDB    000032 RG           #7-106       7-153
$NIRTN    000026 RG             7-104     #7-105
$NIXPT    000002 RG           #7-98       *13-586      *13-604
$NTLPT  = ******    GX          8-234
```

```
NTITIM11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:33     PAGE 2

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

CALL               7-96        7-114       7-117
CALLR              8-152
CLKDF$            #6-60        6-63
MTPS               7-98        7-111
NHWDF$            #6-60        6-62
RETURN             7-119       8-153       9-194
SWSTK$             7-96
```

```
                                .TITLE  CFECOM - GET A COMMAND LINE
                                .IDENT  /V05.00/

                        ;
                        ; COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                        ; TRANSFERRED.
                        ;
                        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                        ; CORPORATION.
                        ;
                        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                        ;
                        ;
                        ; MODULE DESCRIPTION:
                        ;
                        ; CFE - GET A COMMAND LINE
                        ;
                        ;
                        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                        ;
                        ; IDENT HISTORY:
                        ;
                        ; 1.00  14-DEC-79
                        ;       DECNET-11M/S V3.0
                        ;       DECNET-11M-PLUS V1.0
                        ;
                        ; 3.00  16-APR-82
                        ;       DECNET-11M V3.1
                        ;       DECNET-11M-PLUS V1.1
                        ;
                        ; 4.00  07-NOV-83
                        ;       DECNET-11M V4.0
                        ;       DECNET-11M-PLUS V2.0
                        ;
                        ; 5.00  22-JUL-85
                        ;       DECnet-11M/S V4.2
                        ;       DECnet-11M-Plus V3.0
                        ;       DECnet-Micro/RSX V1.0
                        ;
                        ;
```

```
    122                                                .SBTTL  $CFDEF - PROCESS DEFINE COMMAND
    123                                        ;+
    124                                        ; *** - $CFDEF - PROCESS DEFINE COMMAND
    125                                        ;
    126                                        ; INPUT:
    127                                        ;       $COMP = THE COMPONENT CODES
    128                                        ;
    129                                        ; OUTPUT:
    130                                        ;       CONTROL IS PASSED TO THE PROPER ROUTINE
    131                                        ;
    132                                        ;-
    133
    134 000252                                 $CFDEF::
    135 000252  016700  000000G                        MOV     $COMP,R0        ; GET THE COMPONENT CODE
    136 000256  006300                                 ASL     R0              ; CONVERT TO A WORD INDEX
    137 000260  000170  177776'                        JMP     @DSPTBL-2(R0)   ; DISPATCH TO PROPER ROUTINE
```

```
CFEDEF      CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 3

MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME        REFERENCES

CALLR             #4-83      8-317
EPRINT            #4-73
ERROR$            #4-62      6-230      8-321      8-324      8-327      12-448
PRINT             #4-57
```

```
Symbol table

DATSET  000160R         S.ANM2= 000005          S.FTYP= 000002          $HDFNB  000156RG        $HLUN2= 000002 G
DEVA    000174R         S.FDB = 000140          S.NFEN= 000020          $HDSP   000150RG        $HMOD   000000RG
DEVL  = 000005          S.FNAM= 000006          UICA    000201R         $HEFN = 000002 G        $HNAME  000152RG
FILA    000206R         S.FNB = 000036          UICL  = 000005          $HELP = ****** GX       $$$   = 000003
FILL  = 000012          S.FNBW= 000017          $CFHLP  000220RG        $HFDB   000004RG        ...PC1= 000010R
N     = 000003          S.FNTY= 000004          $CMDPT= ****** GX       $HLUN1= 000006 G        ...TPC= 000140
S.ANM1= 000014

. ABS.  000000     000   (RW,I,GBL,ABS,OVR)
        000230     001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 9920 Words  ( 39 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:08.33
SY:CFEHLP.V2,[132,134]CFEHLP/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]CFEHLP
```

```
                                     .SBTTL  BIT DEFINITIONS AND FUNCTION CODES
                             ;
                             ; BIT DEFINITIONS AND FUNCTION CODES
                             ;

                             ;*****
                             ; LINE DESCRIPTOR BLOCK
                             ;
000000                               .ASECT
              000000                 .=0
000000                       L$NAM:: .BLKW   1               ; DEVICE NAME
000002                       L$CNT:: .BLKW   1               ; CONTROLLER NUMBER
000004                       L$UNT:: .BLKW   1               ; UNIT NUMBER
000006                       L$TRI:: .BLKW   1               ; TRIBUTARY NUMBER
002054                               .PSECT
```

K 1

```
**FILE**ID**NTIX25







NN      NN  TTTTTTTTTT    IIIIII    XX      XX    222222    5555555555
NN      NN  TTTTTTTTTT    IIIIII    XX      XX    222222    5555555555
NN      NN      TT          II      XX      XX  22      22  55
NN      NN      TT          II      XX      XX  22      22  55
NNNN    NN      TT          II        XX  XX            22  555555
NNNN    NN      TT          II        XX  XX            22  555555
NN  NN  NN      TT          II          XX            22          55
NN  NN  NN      TT          II          XX            22          55
NN    NNNN      TT          II        XX  XX      22              55
NN    NNNN      TT          II        XX  XX      22              55
NN      NN      TT          II      XX      XX  22        55      55      ....
NN      NN      TT          II      XX      XX  22        55      55      ....
NN      NN      TT        IIIIII    XX      XX  2222222222    555555      ....
NN      NN      TT        IIIIII    XX      XX  2222222222    555555      ....




LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

L 1

```
NTIX25 - NTINIT 'MODULE X25-SER MACRO V05.03b Saturday 29-Jun-85 03:29
```

K 2

```
**FILE**ID**NTICEX




NN      NN   TTTTTTTTTT      IIIIII       CCCCCCCC   EEEEEEEEEE   XX      XX
NN      NN   TTTTTTTTTT      IIIIII       CCCCCCCC   EEEEEEEEEE   XX      XX
NN      NN       TT            II       CC           EE           XX      XX
NN      NN       TT            II       CC           EE           XX      XX
NNNN    NN       TT            II       CC           EE             XX  XX
NNNN    NN       TT            II       CC           EE             XX  XX
NN  NN  NN       TT            II       CC           EEEEEEEE         XX
NN  NN  NN       TT            II       CC           EEEEEEEE         XX
NN    NNNN       TT            II       CC           EE             XX  XX
NN    NNNN       TT            II       CC           EE             XX  XX
NN      NN       TT            II       CC           EE           XX      XX        ....
NN      NN       TT            II       CC           EE           XX      XX        ....
NN      NN       TT          IIIIII       CCCCCCCC   EEEEEEEEEE   XX      XX        ....
NN      NN       TT          IIIIII       CCCCCCCC   EEEEEEEEEE   XX      XX        ....



     11           11         SSSSSSSS
     11           11         SSSSSSSS
   1111         1111       SS
   1111         1111       SS
     11           11       SS
     11           11       SS
     11           11         SSSSSS
     11           11         SSSSSS
     11           11               SS
     11           11               SS
     11           11               SS
     11           11               SS
   111111       111111     SSSSSSSS
   111111       111111     SSSSSSSS
```

L 2

```
NTICEX  MACRO V05.03b Saturday 29-Jun-85 03:29
```

```
                                                                     K  3
NTICEX11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 3
SYMBOL CROSS REFERENCE                                   CREF    04.00
SYMBOL  VALUE                REFERENCES

ZF.DDM  = 000001             #6-75
ZF.DIA  = 004000             #6-75
ZF.DLC  = 000002             #6-75
ZF.DVP  = 100000             #6-75
ZF.INI  = 040000             #6-75         8-193
ZF.KMX  = 000020             #6-75
ZF.LLC  = 000004             #6-75
ZF.LMC  = 000100             #6-75
ZF.MAN  = 020000             #6-75
ZF.MFL  = 000010             #6-75
ZF.MTM  = 000400             #6-75
ZF.MUX  = 000040             #6-75
ZF.PSE  = 002000             #6-75
ZF.SLI  = 010000             #6-75
ZF.TIM  = 000200             #6-75
ZF.X3P  = 000000             #6-75
ZS.ASN  = 100000             #6-75
ZS.BSY  = 140000             #6-75
Z.AVL     000014             #6-75
Z.DAT     000016             #6-75         8-197
Z.DSP     000000             #6-75         6-75
Z.FLG     000010             #6-75         8-193
Z.LEN   = 000016             #6-75
Z.LLN     000006             #6-75
Z.MAP     000020             #6-75
Z.NAM     000004             #6-75         8-203
Z.PCB     000012             #6-75         8-195
Z.SCH     000007             #6-75
$NICEX    000000 RG          #7-135
$NICSR  = ******   GX         8-268
$NIDLI  = ******   GX         8-217
$NITIM  = ******   GX         8-279
$NIUMR  = ******   GX         8-250
.BASEB  = 140000             #6-74         8-231         8-234
.CXCSR  = ******   GX         8-234
```

```
                                                                     L  3
NTICEX11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 4
```

```
**FILE**ID**NTIDLI



NN        NN    TTTTTTTTTT         IIIIII        DDDDDDDD        LL                 IIIIII
NN        NN    TTTTTTTTTT         IIIIII        DDDDDDDD        LL                 IIIIII
NN        NN         TT              II          DD        DD    LL                   II
NN        NN         TT              II          DD        DD    LL                   II
NNNN      NN         TT              II          DD        DD    LL                   II
NNNN      NN         TT              II          DD        DD    LL                   II
NN   NN   NN         TT              II          DD        DD    LL                   II
NN   NN   NN         TT              II          DD        DD    LL                   II
NN      NNNN         TT              II          DD        DD    LL                   II
NN      NNNN         TT              II          DD        DD    LL                   II
NN        NN         TT              II          DD        DD    LL                   II              ....
NN        NN         TT              II          DD        DD    LL                   II              ....
NN        NN         TT            IIIIII        DDDDDDDD        LLLLLLLLLL         IIIIII            ....
NN        NN         TT            IIIIII        DDDDDDDD        LLLLLLLLLL         IIIIII            ....



     11              11            SSSSSSSS
     11              11            SSSSSSSS
   1111            1111          SS
   1111            1111          SS
     11              11          SS
     11              11          SS
     11              11            SSSSSS
     11              11            SSSSSS
     11              11                    SS
     11              11                    SS
     11              11                    SS
     11              11                    SS
   111111          111111        SSSSSSSS
   111111          111111        SSSSSSSS
```

```
                                                            K 5
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:31   Page 7
LOCAL DATA

     81                                             .SBTTL  LOCAL DATA
     82
     83                                     ;
     84                                     ; CONSTANT DEFINITIONS
     85                                     ;
     86              176000                 AREAMK  = 176000                       ; AREA BITS MASK
     87              002000                 LVL1    = 2000                         ; LEVEL 1 AREA
     88
     89                                     ; LOCAL DATA (IMPURE)
     90                                     ;
     91 000000                                      .PSECT  DATA,D
     92
     93                                     ;
     94                                     ; LOCAL STORAGE
     95                                     ;
     96                                             .IF NDF R$$11S
     97                                     PLDADR: .BLKW   1                      ; PLD DATA BASE ADDRESS
     98                                     PLDLEN: .BLKW   1                      ; LENGTH OF THE PLD DATA BASE
     99                                             .ENDC   ; NDF R$$11S
    100
    101 000000   070410   143074           RCP1NM: .RAD50  /RCP1../               ; ROUTING CONTROL PROGRAM (LEVEL 1)
    102 000004   070410   146174           RCP2NM: .RAD50  /RCP2../               ; ROUTING CONTROL PROGRAM (LEVEL 2)
    103
    104 000010                              LONACP: RUN$    NETACP,,,,,1,1         ;DIRECTIVE TO REQUEST NETACP
    105 000036                              ASSIGN: ALUN$   2,NS,0                 ;ASSIGN LUN 2 TO NS:
```

```
                                                            L 5
NTILON - NTINIT 'EXECUTOR ON' R MACRO V05.03b Saturday 29-Jun-85 03:31   Page 8
```

```
                                                                          K  6
NTILON11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:31      PAGE 3
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

ZF.DDM   = 000001           #6-70
ZF.DIA   = 004000           #6-70
ZF.DLC   = 000002           #6-70
ZF.DVP   = 100000           #6-70
ZF.INI   = 040000           #6-70
ZF.KMX   = 000020           #6-70
ZF.LLC   = 000004           #6-70
ZF.LMC   = 000100           #6-70
ZF.MAN   = 020000           #6-70
ZF.MFL   = 000010           #6-70
ZF.MTM   = 000400           #6-70
ZF.MUX   = 000040           #6-70
ZF.PSE   = 002000           #6-70
ZF.SLI   = 010000           #6-70
ZF.TIM   = 000200           #6-70
ZF.X3P   = 000000           #6-70
ZS.ASN   = 100000           #6-70
ZS.BSY   = 140000           #6-70
Z.AVL      000014           #6-70
Z.DAT      000016           #6-70
Z.DSP      000000           #6-70      6-70
Z.FLG      000010           #6-70
Z.LEN    = 000016           #6-70
Z.LLN      000006           #6-70      9-158
Z.MAP      000020           #6-70
Z.NAM      000004           #6-70
Z.PCB      000012           #6-70
Z.SCH      000007           #6-70
$DBECL   = ******   GX        9-144    9-270
$DBXPT   = ******   GX        9-142    9-196
$ERR11     000046 RG        #8-112     9-297
$ERR15     000136 RG        #8-113     9-152      9-192      9-294
$ERR17     000176 RG        #8-114     9-295
$ERR18     000252 R         #8-115     9-296
$ERR19     000312 RG        #8-116     9-195
$NIERR   = ******   GX        8-112    8-113      8-114      8-115      8-116
$NILON     000000 RG        #9-140
$NIMFE   = ******   GX        9-276
$NIXPT   = ******   GX        9-154
$$$GLB   = ******             7-104    7-105
$$$T1    = 000002           #7-104     7-104      7-104      7-104     #7-105
$$$T2    = 000042 R         #7-105     7-105
.BASEB   = 140000           #6-66
```

```
                                                                          L  6
NTILON11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:31      PAGE 4
```

```
                                                                 K  7
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:52  Page 9-3
Symbol table

LF.MDC= 000100          NF$SCN= 000020          N$$NCT= 000001          P$OCCB  000050          S$TIM   000014
LF.MFL= 004000          NF$SHU= 000004          N$$PEM= 000001          P$PFQ   000006          S$TMR   000016
LF.MTP= 000020          NF$TIM= 000200          PDVTA = ****** GX       P$PKSZ  000044          S$WND   000006
LF.PAC= 000200          N$ACQ   000000          PF$BLK= 020000          P$RMX1  000016          S$$WRG= 000000
LF.RDY= 040000          N$ACTL  000032          PF$CLC= 010000          P$RMX2  000020          S$$YSZ= 007600
LF.REA= 010000          N$ADJ1  000072          PF$DLM= 100000          P$RPRI  000042          S.COST  000001
LF.SER= 000040          N$ADJ2  000074          PF$EIP= 000002          P$RTIM  000003          S.FLG   000000
LF.TIM= 000010          N$CACH  000062          PF$ENB= 000001          P$STA1  000022          S.LEN   000004
LF.UNL= 020000          N$CIR   000034          PF$FAI= 004000          P$STA2  000023          S.NMST  000002
LF.X2P= 000000          N$CRC   000120          PF$FM1= 000100          P$TIM   000010          S.OWNR  000003
LLCRS = ****** GX       N$DLA   000020          PF$FM2= 000200          P$TSCT  000026          TMEFN = 000001
LLCTA = ****** GX       N$DLY   000014          PF$OFF= 000000          P$TSIZ  000024          T$FLAG  000044
LN.CLO= 000000          N$ELEN  000054          PF$RM1= 000020          P$TYP   000001          T$LIF   000013
LN.DUM= 000005          N$ENC   000042          PF$RM2= 000040          P$$P45= 000000          T$LIFL  000013
LN.LOA= 000004          N$ERRC  000022          PF$RVR= 000010          P$$WRD= 000000          T$LIFO  000013
LN.LOO= 000003          N$FLG   000005          PF$STA= 000004          Q$$OPT= 000010          T$LIFS  000013
LN.OAU= 000003          N$FNC   000006          PF$SVC= 040000          RF.CTL= 000003          T$LIN   000000
LN.OFF= 000001          N$GENQ  000052          PF$UP = 000004          RF.LD1= 000040          T$LIPS  000006
LN.ON = 000000          N$GTM   000015          PR$BED= 000200          RF.LD2= 000100          T$LLD   000012
LN.OOP= 000004          N$HC1   000052          PR$BEU= 000100          RF.TIM= 177400          T$LLDC  000045
LN.OPE= 000001          N$HC2   000056          PR$BRD= 000040          RF.TMO= 000400          T$LLDL  000012
LN.REF= 000002          N$HIGH  000033          PR$BRU= 000020          RF.WFC= 000200          T$LLDO  000012
LN.SER= 000002          N$LLT   000026          PR$DWN= 000002          RF.WTD= 000020          T$LLDS  000012
LN.STA= 000017          N$LLTM  000024          PR$LCC= 000010          RF.WTM= 000030          T$LLEN  000046
LN.SUB= 000360          N$LVC   000036          PR$MOP= 000004          RF.WTS= 000010          T$LOPR  000002
LN.TRI= 000006          N$LV1   000002          PR$UP = 000001          R$$DER= 000000          T$LTCL  000024
L$$ASG= 000000          N$LV2   000010          PS$CHR= 000016          R$$K11= 000001          T$LTIM  000026
L$$DRV= 000000          N$MBXQ  000050          PS$FAI= 000014          R$$SND= 000000          T$LTPR  000014
L$$P11= 000001          N$MHC1  000036          PS$NTI= 000006          R$$11M= 000000          T$LTPS  000020
L$$11R= 000000          N$MHC2  000044          PS$OFF= 000000          R$$11S= 000001          T$NAPL  000004
L.COST  000015          N$PLD   000016          PS$STR= 000002          SF$BLK= 000001          T$NFE   000000
L.CTL   000012          N$PLLT  000030          PS$UP = 000012          SF$INC= 000002          T$NLEN  000010
L.CVA   177776          N$PRI   000076          PS$VER= 000010          SF.ACT= 000200          T$NNUL  000002
L.DDM   000002          N$ROA1  000022          PS$WT = 000004          SF.ENA= 000100          T$NOPL  000006
L.DDS   000004          N$ROA2  000030          PT$BRO= 000200          SF.LPB= 000004          T$NRNI  000042
L.DLC   000003          N$RTMX  000014          PT$DRT= 000100          SF.MFL= 000040          T$NRPL  000005
L.DLM   000006          N$RTM1  000014          PT$END= 000004          SF.PAC= 000020          T$NRUL  000007
L.DLS   000010          N$RTM2  000015          PT$LV1= 000002          SF.REA= 000010          T$NVR   000001
L.FLG   000000          N$RT1   000000          PT$LV2= 000001          SF.SER= 000001          T$RPRI  000040
L.KRBA  000016          N$RT2   000006          PT$PH3= 000010          SF.SVC= 000002          T$SVC   000034
L.LEN = 000022          N$SLA   000016          PT$XAR= 000020          SF.UNL= 000040          T$T5    000030
L.MPF   000022          N$SNOD  000012          PX$BLK= 000040          SLTMA = ****** GX       T$T6    000032
L.NMST  000020          N$TCTL  000112          PX$DLM= 000200          SLTNM = ****** GX       T$$KMG= 000000
L.NSTA  000014          N$TIM   000004          PX$SVC= 000100          STAADD  000000R         T$$MIN= 000000
L.OWNR  000021          N$TLC   000100          P$CHN   000004          STANUM  000002R         VF$RCV= 100000
L.UNT   000013          N$TNC   000114          P$CNT   000005          STRT  = 000004          VF$XMT= 040000
M$$CRB= 000124          N$TRC   000106          P$CTR   000034          S$ADSZ  000007          V$FLG   000000
M$$CRX= 000000          N$TTCB  000110          P$DRCT  000015          S$CTL   000002          V$LEN   000022
M$$EXT= 000000          N$VCB   000010          P$DRTR  000036          S$DTE   000020          V$RCV   000002
M$$FCS= 000000          N$VER   000066          P$FLG   000012          S$FLG   000010          V$XMT   000012
M$$MGE= 000000          N$XLEN  000124          P$FWD   000030          S$LEN   000032          V$$CTR= 001000
M$$NET= 000000          N$$ACC= 000001          P$ICCB  000046          S$LFT   000012          X$$DBT= 000000
M$$OVR= 000000          N$$BUF= 000001          P$IPL   000014          S$LNK   000000          ZF.COU= 001000
NF$BLK= 000100          N$$LDV= 000001          P$LCD   000002          S$OWNR  000030          ZF.DDM= 000001
NF$DMO= 000010          N$$MCP= 000001          P$LEN   000052          S$PKSZ  000004          ZF.DIA= 004000
NF$MOU= 000040          N$$MLL= 000001          P$LST   000000          S$RCL   000013          ZF.DLC= 000002
NF$RST= 000002          N$$MOV= 000010          P$NRNI  000040          S$STA   000003          ZF.DVP= 100000
```

```
                                                                 L  7
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:52  Page 9-4
```

```
    159                                                 .SBTTL  ERROR MESSAGES
    160
    161                                         ;
    162                                         ; ERROR MESSAGE BUFFER
    163                                         ;
    164 000124     015      116      124        NTIERN: .ASCII  <15>/NTINIT -- /        ;MESSAGE PREFIX (NAME OF NTINIT)
        000127     111      116      111
        000132     124      040      055
        000135     055      040
    165 000137                                  NTIERB: .BLKB   70.                     ;MESSAGE BODY
    166                                                 .EVEN
    167
    168                                         ;
    169                                         ; ERROR MESSAGES
    170                                         ;
    171                                         ; MESSAGES IN THIS MODULE SHOULD BE NUMBERED 01 TO 09
    172                                         ; MESSAGES 10-11 ARE IN NTICEX
    173                                         ; MESSAGES 11-19 ARE IN NTILON
    174                                         ; MESSAGES 20-24 ARE IN NTICSR
    175                                         ; MESSAGES 25-29 ARE IN NTIKPN OR NTIKLC
    176                                         ; MESSAGES 30-34 ARE IN NTIMFL
    177                                         ; MESSAGES 35-39 ARE IN NTIMFE
    178                                         ; MESSAGES 40-44 ARE IN NTIX25
    179                                         ;
    180                                         ; THE MAIN ERROR PROCESSOR IS "$NIERR"
    181                                         ; ASTERISK REPLACERS ARE:
    182                                         ;   $NER50 - INSERT RAD50 STRING FROM $NIERA
    183                                         ;   $NERDN - INSERT RAD50 STRING FROM $NIERA AND DECIMAL NUMBER FROM $NIERA+2
    184                                         ;   $NERDD - INSERT RAD50 STRING FROM $NIERA, DECIMAL NUMBER FROM $NIERA+2,
    185                                         ;               AND DECIMAL NUMBER FROM $NIERA+4
    186                                         ; SECOND LINE FORMATTING AND PRINTING IS NOT SUPPORTED
    187                                         ;
    188                                                 .ENABL  LC
    189                                                 .IF NDF R$$11S
    190                                         NTLER$  ,02,NONE,$NIERR,,,<Network Loader (NTL...) Not Installed>
    191                                                 .ENDC
    192 000246                                  NTLER$  ,03,NONE,$NIERR,,,<Illegal Function>
    193 000300                                  NTLER$  ,04,NONE,$NIERR,,,<Comm Exec Not Loaded>
    194 000336                                  NTLER$  <:>,12,NONE,$NIERR,,,<Line in Wrong State, Only First Line Enabled>
    195
    196                                                 .DSABL  LC
    197
    198                                         ;+
    199                                         ; VECTORED EXEC TABLE
    200                                         ;
    201
    202 000424  000000                          EXEVEC: .WORD   0
    203 000426  000000G                         FMASK:: $FMASK                          ; RSX-11M+ VECTORED SYMBOLS
    204 000430  172354                          KSAR6:: KISAR6
    205
    206                                                 .IF DF  R$$MPL
    207                                         HFMSK:: $HFMSK
    208                                         SAHDB:: $SAHDB
    209                                         SAHPT:: $SAHPT
    210                                         URMST:: $URMST
    211                                         NCPU::  $NCPU
    212                                                 .ENDC
```

```
                                ;       .SBTTL  ERROR SUBROUTINES

                                ;+
                                ; $NER50 - INSERT A RAD50 STRING
                                ; $NERDN - INSERT A RAD50 STRING AND A DECIMAL NUMBER
                                ; $NERDD - INSERT A RAD50 STRING AND TWO DECIMAL NUMBERS
                                ;-

                                        .IF NDF S$$BAS

                                ;
                                ; $NIERA        = RAD50 WORD
                                ;
000504  016701  000010'         $NER50::MOV     $NIERA,R1               ;GET RAD50 NAME
000510                                  CALL    $C5TA                   ;INSERT RAD50
000514  122740  000040          10$:    CMPB    #' ,-(R0)               ;SKIP TRAILING BLANKS
000520  001775                          BEQ     10$                     ;
000522  005200                          INC     R0                      ;ADJUST FOR AUTO-DECREMENT
000524                                  RETURN

                                ;
                                ; $NIERA        = RAD50 WORD
                                ; $NIERA+2      = DECIMAL NUMBER
                                ;
000526                          $NERDN::CALL    $NER50                  ;INSERT RAD50 STRING
000532  112720  000055                  MOVB    #'-,(R0)+               ;PUT IN THE SEPARATOR
000536  016701  000012'                 MOV     $NIERA+2,R1             ;GET OCTAL NUMBER
000542  005002                          CLR     R2                      ;ZERO SUPPRESS
000544                                  CALLR   $CBDMG                  ;PUT IN DECIMAL NUMBER

                                ;
                                ; $NIERA        = RAD50 WORD
                                ; $NIERA+2      = DECIMAL NUMBER
                                ; $NIERA+4      = DECIMAL NUMBER (IF MI, DON'T PRINT)
                                ;
000550                          $NERDD::CALL    $NERDN                  ;PUT IN RAD50 AND FIRST OCTAL
000554  016701  000014'                 MOV     $NIERA+4,R1             ;GET SECOND NUMBER
000560  100405                          BMI     10$                     ;IF MI, DON'T PRINT
000562  112720  000055                  MOVB    #'-,(R0)+               ;PUT IN SECOND SEPARATOR
000566  005002                          CLR     R2                      ;ZERO SUPPRESS
000570                                  CALLR   $CBDMG                  ;PUT IN DECIMAL NUMBER
000574                          10$:    RETURN

                                        .ENDC
```

```
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE            REFERENCES

$PDVID  = ******  GX      8-247
$PDVNM  = ******  GX      8-246
$PDVTA  = ******  GX      8-250
$PSIPT  = ******  GX      8-241
$PUMR   = ******  GX      8-218
$QASTT  = ******  GX      8-221
$SLTMA  = ******  GX      8-232
$SLTNM  = ******  GX      8-233
$SRSTD  = ******  GX      8-215
$TKPS   = ******  GX      8-223
$TKTCB  = ******  GX      8-220
$TK100  = ******  GX      8-238
$T1SCL  = ******  GX      8-243
$T100C  = ******  GX      8-236
$T100Q  = ******  GX      8-237
$UMRPT  = ******  GX      8-214
$$$ARG  = 000002         #14-689    14-689   #14-689    14-689   #14-689    14-689   #14-689    14-689    14-689
                         #14-689    14-689    14-689    14-689    14-689    14-689    14-689    14-689    14-689
                          14-689    14-689   #14-689    14-689    14-689    14-689    14-689    14-689    14-689
                          14-689    14-689    14-689    14-689   #14-689    14-689    14-689    14-689    14-689
                          14-689    14-689    14-689    14-689    14-689    14-689
$$$GLB  = ******          7-104     7-109     7-110
$$$T1   = 000006         #7-104     7-104    #7-110     7-110     7-110     7-110
$$$$    = 000001         #18-846   #18-847    18-851
.BASEB  = 140000         #6-77
.CXALL    000062  G      #6-72
.CXCSR    000004  G      #6-72
.CXLBR    000060  G      #6-72
.CXLB1    000030  G      #6-72
.CXLB2    000044  G      #6-72
.CXLEN    000064  G      #6-72
.CXPCB    000006  G      #6-72
.CXSYM    000010  G      #6-72
.CXUNL    000024  G      #6-72
.CXVEC    000026  G      #6-72
.CXVSM    000016  G      #6-72
.NTGCD    000000         #6-72
.NTPCB    000000  G      #6-72
.NTUMR    000002  G      #6-72
.SBPCB    000066  G      #6-72
```

K 11

```
**FILE**ID**NTIUMR




NN      NN      TTTTTTTTTT        IIIIII        UU      UU      MM      MM      RRRRRRRR
NN      NN      TTTTTTTTTT        IIIIII        UU      UU      MM      MM      RRRRRRRR
NN      NN          TT              II          UU      UU      MMMM  MMMM      RR      RR
NN      NN          TT              II          UU      UU      MMMM  MMMM      RR      RR
NNNN    NN          TT              II          UU      UU      MM  MM  MM      RR      RR
NNNN    NN          TT              II          UU      UU      MM  MM  MM      RR      RR
NN  NN  NN          TT              II          UU      UU      MM      MM      RRRRRRRR
NN  NN  NN          TT              II          UU      UU      MM      MM      RRRRRRRR
NN    NNNN          TT              II          UU      UU      MM      MM      RR  RR
NN    NNNN          TT              II          UU      UU      MM      MM      RR  RR
NN      NN          TT              II          UU      UU      MM      MM      RR    RR            ....
NN      NN          TT              II          UU      UU      MM      MM      RR    RR            ....
NN      NN          TT            IIIIII        UUUUUUUUUU      MM      MM      RR      RR          ....
NN      NN          TT            IIIIII        UUUUUUUUUU      MM      MM      RR      RR          ....



     11              11           SSSSSSSS
     11              11           SSSSSSSS
   1111            1111         SS
   1111            1111         SS
     11              11         SS
     11              11         SS
     11              11           SSSSSS
     11              11           SSSSSS
     11              11                 SS
     11              11                 SS
     11              11                 SS
     11              11                 SS
   111111          111111       SSSSSSSS
   111111          111111       SSSSSSSS
```

L 11

```
                                        ;****
                                        ; LOCAL MACROS
                                        ;****

                                                .MACRO  PRINT   TEXT
                                                MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
                                                CALL    $PRINT          ; PRINT MESSAGE
                                                .ENDM   PRINT

                                                .MACRO  ERROR$  TEXT
                                        ;
                                        ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
                                        ; MESSAGE STRING.
                                        ;
                                                .IF DIF <TEXT><R0>
                                                MOV     TEXT,R0                 ; GET ADDRESS OF ERROR MESSAGE
                                                .ENDC
                                                CALL    $CFERR                  ; PRINT ERROR MESSAGE
                                                .ENDM   ERROR$

                                                .MACRO  EPRINT  TEXT
                                        ;
                                        ; PRINT TEXT ON ERROR LUN
                                        ;
                                                .IF DIF <TEXT><R0>
                                                MOV     TEXT,R0                 ; GET ADDRESS OF ASCIZ STRING TO PRINT
                                                .ENDC
                                                CALL    $CFER1                  ; PRINT MESSAGE ON ERROR LUN
                                                .ENDM   EPRINT
                                        ;****
                                        ; LOCAL DATA
                                        ;****

                                        ;
                                        ; ERROR MESSAGES
                                        ;
                                                .NLIST  BEX
                                                .ENABL  LC
        000002                                  EX$ERR = 2
000000     002       103       106      ERRLID::.ASCIZ  <EX$ERR>/CFE -- Invalid line identification/
000044     002       103       106      ERRCID::.ASCIZ  <EX$ERR>/CFE -- Invalid circuit identification/
000113     002       103       106      ERRPRO::.ASCIZ  <EX$ERR>/CFE -- Invalid process identification/
000162     002       103       106      ERRCHN::.ASCIZ  <EX$ERR>/CFE -- Error in channel list/
000220     002       103       106      ERRLTY::.ASCIZ  <EX$ERR>/CFE -- Operation invalid for line type/
000270     002       103       106      ERRCAL::.ASCIZ  <EX$ERR>/CFE -- Error in call mask or call value/
000341     002       103       106      ERRWLD::.ASCIZ  <EX$ERR>/CFE -- Invalid wildcard syntax/
000401     002       103       106      ERRFSP::.ASCIZ  <EX$ERR>/CFE -- Error in filespec/
000433     002       103       106      ERRIPV::.ASCIZ  <EX$ERR>/CFE -- Invalid parameter value/
000473     012       040       040      CRLF:   .ASCII  <12>"      "<200>           ; HEADER FOR BAD COMMAND LINE
000501     015       200                CR:     .ASCII  <15><200>               ; SINGLE CARRAIGE RETURN
000503     076       076       200      ARROWS: .ASCII  ">>"<200>               ; MARKER FOR BAD PART OF COMMAND LINE
                                                .EVEN
000506                                  $ERROR::.BLKW   1                       ; ADDRESS OF ERROR MESSAGE
                                                                                ; RETURN FROM ACTION ROUTINES IF C SET
                                        .SBTTL  SETEXE - SET EXECUTOR PARAMETERS
                                ;+
                                ; *** - SETEXE - SET EXECUTOR PARAMETERS
                                ;
                                ; INPUTS:
                                        $OPRND = OPERANDS WORD
                                ;
                                ; OUTPUT:
                                        CONTROL DISPATCHED TO PROPER ROUTINE
                                ;
                                ;-

000264                          SETEXE:
000264  032767  000000G 000000G         BIT     #OE.ROU,$OPRND  ; NEED TO MODIFY THE ROUTING PARAMETERS ?
000272  001004                          BNE     5$              ; IF NE, YES
000274  032767  000000G 000000G         BIT     #E2.ROU,$OPRN2  ; REALLY NO CHANGES ?
000302  001403                          BEQ     10$             ; IF EQ, NO
000304  004767  000000G         5$:     CALL    $SEROU          ; CHANGE THE ROUTING PARAMETERS
000310  103504                          BCS     90$             ; IF CS, END NODE - ABORT COMMAND

000312  032767  000000G 000000G 10$:    BIT     #OE.NSP,$OPRND  ; DID ANY NSP PARAMETERS CHANGE ?
000320  001004                          BNE     15$             ; IF NE, YES
000322  032767  000000G 000000G         BIT     #E2.NSP,$OPRN2  ; REALLY NONE ?
000330  001402                          BEQ     20$             ; IF EQ NO NSP PARAMETERS
000332  004767  000000G         15$:    CALL    $SENSP          ; MODIFY NSP

000336  032767  000000G 000000G 20$:    BIT     #OE.NOD,$OPRND  ; NEED TO MODIFY THE NODE DEFINITION ?
000344  001402                          BEQ     30$             ; IF EQ, NO
000346  004767  000000G                 CALL    $SENOD          ; ELSE, DO IT

000352  032767  000000G 000000G 30$:    BIT     #OE.SUB,$OPRND  ; TRANSPORT SUBADDRESS SPECIFIED ?
000360  001460                          BEQ     90$             ; IF EQ, NOPE
                                ;
                                ; SET EXECUTOR TRANSPORT SUBADDRESSES:
                                ;  IF A RANGE IS SPECIFIED AND IS NOT 0-0 SET IT UP
                                ;  IF A RANGE IS SPECIFIED AND IS 0-0 CLEAR BOTH SUBADDRESSES
                                ;  IF ONE VALUE IS SPECIFIED:
                                ;   IF THERE ARE NO SUBADDRESSES IN TEMPLATE SETUP LOW SUBADDRESS
                                ;   IF ONE SUBADDRESS IS PRESENT IN TEMPLATE UPDATE THAT SUBADDRESS
                                ;   IF BOTH SUBADDRESSES ARE PRESENT:
                                ;       IF VALUE IS < LOW SUBADDRESS,  UPDATE LOW SUBADDRESS
                                ;       IF VALUE IS > LOW SUBADDRESS, UPDATE HIGH SUBADDRESS

000362  016700  000000G                 MOV     $PSIPT,R0       ; POINT TO PSN$DF
000366  001456                          BEQ     100$            ; IF EQ, NO PSN$DF - ERROR
000370  016046  000000G                 MOV     C.STS(R0),-(SP) ; GET FLAGS
000374  042716  000000C                 BIC     #^C<CS.THI!CS.TLO>,(SP) ; CLEAR DON'T CARE BITS
000400  016701  000000G                 MOV     $ESUB,R1        ; GET LOW VALUE
000404  016702  000002G                 MOV     $ESUB+2,R2      ; GET HI VALUE
000410  100015                          BPL     50$             ; IF PL, USER SPECIFIED HIGH AND LOW
                                ;
                                ; ONE VALUE SPECIFIED ON COMMAND LINE. IF ONLY ONE SUBADDRESS IN TEMPLATE
                                ; UPDATE THAT ONE, ELSE FIND THE BEST FIT AND UPDATE THAT SUBADDRESS.
                                ;
000412  022716  000000G                 CMP     #CS.THI,(SP)    ; ONLY HIGH PRESENT ?
000416  001410                          BEQ     40$             ; IF EQ, YES
000420  022716  000000G                 CMP     #CS.TLO,(SP)    ; ONLY LOW PRESENT ?
```

```
**FILE**ID**CFEDSP



  CCCCCCCC   FFFFFFFFFF   EEEEEEEEEE   DDDDDDDD       SSSSSSSS   PPPPPPPP
  CCCCCCCC   FFFFFFFFFF   EEEEEEEEEE   DDDDDDDD       SSSSSSSS   PPPPPPPP
CC           FF           EE           DD      DD   SS           PP      PP
CC           FF           EE           DD      DD   SS           PP      PP
CC           FF           EE           DD      DD   SS           PP      PP
CC           FF           EE           DD      DD   SS           PP      PP
CC           FFFFFFFF     EEEEEEEE     DD      DD     SSSSSS     PPPPPPPP
CC           FFFFFFFF     EEEEEEEE     DD      DD     SSSSSS     PPPPPPPP
CC           FF           EE           DD      DD           SS   PP
CC           FF           EE           DD      DD           SS   PP
CC           FF           EE           DD      DD           SS   PP              ....
CC           FF           EE           DD      DD           SS   PP              ....
  CCCCCCCC   FF           EEEEEEEEEE   DDDDDDDD     SSSSSSSS     PP              ....
  CCCCCCCC   FF           EEEEEEEEEE   DDDDDDDD     SSSSSSSS     PP              ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
CFEHLP     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE                REFERENCES

DATSET     000160 R            4-88     #4-90
DEVA       000174 R            4-90     #4-92      4-93
DEVL     = 000005              4-90     #4-93
FILA       000206 R            4-90     #4-98      4-99
FILL     = 000012              4-90     #4-99
N        = 000003             #4-88      4-88
S.FDB    = 000140              4-88      4-88
UICA       000201 R            4-90     #4-95      4-96
UICL     = 000005              4-90     #4-96
$CFHLP     000220 RG          #5-115
$CMDPT   = ******  GX          5-116
$HDFNB     000156 RG          #4-88
$HDSP      000150 RG          #4-88
$HEFN    = 000002  G          #4-88
$HELP    = ******  GX          5-117
$HFDB      000004 RG          #4-88
$HLUN1   = 000006  G          #4-88
$HLUN2   = 000002  G          #4-88
$HMOD      000000 RG          #4-88
$HNAME     000152 RG          #4-88
$$$      = 000003             #4-88      4-88
...PC1   = 000010 R           #4-88
```

```
                                          .SBTTL  DEFINE TEMPLATES GLOBALLY

                                  ;*****
                                  ; DEFINE THE SLT TEMPLATE
                                  ;
 000010                                   .ASECT
            000000                        .=0
 000000                                   .BLKW   1                       ; C.LNK
 000002                                   .BLKW   1                       ; C.STS
 000004                           S$LNAM::.BLKW   1                       ; THE DDM PROCESS ASSOCIATED WITH THIS  LINE
 000006                           S$LFLG::.BLKW   1                       ; FLAGS WORD
 000010                           S$LCNT::.BLKB   1                       ; CONTROLLER NUMBER OF THE ASSOCIATED DDM
 000011                           S$LUNT::.BLKB   1                       ; THE UNIT NUMBER OF THE ASSOCIATED DDM
 000012                           S$LCTT::.BLKW   1                       ; THE COUNTER TIMER VALUE (LAPB LINES ONLY)
 000014                           S$LTYP::.BLKB   1                       ; 0=SLAVE, 1=MASTER
 000015                                   .BLKB   1                       ; (UNUSED)
 000016                           S$LHTM::.BLKW   1                       ; HELLO TIMER
 000020                           S$LRPR::.BLKW   1                       ; ROUTING PRIORITY
 000022                           S$LBRA::.BLKW   1                       ; BROADCAST ROUTER ADJACENCIES
            000024                S$LLEN==.
 002054                                   .PSECT

                                  ;*****
                                  ; DEFINE DDM TEMPLATE
                                  ;
 000024                                   .ASECT
            000000                        .=0
 000000                                   .BLKW   1                       ; C.LNK
 000002                                   .BLKW   1                       ; C.STS
 000004                           D$DNAM::.BLKW   1                       ; DDM PROCESS NAME
 000006                           D$DFLG::.BLKW   1                       ; DDM FLAGS WORD
 000010                           D$DPRI::.BLKW   1                       ; DDM PROCESS PRIORITY
 000012                           D$DNLT::.BLKB   1                       ; NUMBER OF LINE TABLES IN PROCESS SPACE
 000013                           D$DNCT::.BLKB   1                       ; NUMBER OF CONTROLLER TABLES IN PROCESS SPACE
            000014                D$DLEN==.
 002054                                   .PSECT

                                  ;*****
                                  ;
                                  ;
                                  ; DEFINE CONTROLLER TEMPLATE
                                  ;
 000014                                   .ASECT
            000000                        .=0
 000000                                   .BLKW   1                       ; C$NLNK
 000002                                   .BLKW   1                       ; C$NSTS
 000004                           C$NCNT::.BLKW   1                       ; CONTROLLER NUMBER
 000006                           C$NVEC::.BLKW   1                       ; VECTOR ADDRESS
 000010                           C$NCSR::.BLKW   1                       ; CSR ADDRESS
 000012                           C$NPRI::.BLKW   1                       ; CONTROLLER PRIORITY
 000014                           C$NURM::.BLKW   1                       ; UNIBUS RUN MASK
            000016                C$NLEN==.
 002054                                   .PSECT

                                  ;*****
                                  ;
                                  ;
                                  ; DEFINE UNIT TEMPLATE
                                  ;
```

Table of contents

```
     5-   50     MACRO DEFINITIONS
     6-   68     LOCAL DATA
     7-  105     SET MODULE X25 SERVER
```

```
NTICEX  MACRO V05.03b Saturday 29-Jun-85 03:29
Table of contents
```

```
     6-   53     DATA
     7-  106     SET CEX FUNCTION
     8-  160     LOCAL INITIALIZATION
```

```
                                                                                          L  3
NTICEX11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:30      PAGE 4

MACRO CROSS REFERENCE                                     CREF    04.00

MACRO NAME        REFERENCES

CALL               8-188      8-202      8-204      8-207      8-219      8-227      8-250      8-270      8-279
CCBDF$            #6-65       6-72
CLEF$S            #6-64
DHBDF$            #6-65       6-76
DIR$              #6-64
EMSG$R            #6-64
NHWDF$            #6-65       6-74
NTLER$            #6-64
PDVDF$            #6-65       6-75
RETURN             8-210      8-235      8-281
RQST$             #6-64
SDAT$             #6-64
SLTDF$            #6-65       6-73
SWSTK$             8-202      8-227
WTSE$S            #6-64
```

```
                                                                                          M  3
**FILE**ID**NTICSR
```

Table of contents

6-   55   DATA
7-   77   NODE INFO INITIALIZATION

```
    107                                     .SBTTL  ERROR MESSAGES
    108                              ;
    109                              ; ERROR MESSAGES
    110                              ;
    111                                     .ENABL  LC
    112 000046                       NTLER$  <:>,11,NONE,$NIERR,,,<Router Priorities Data Base Allocation Failure>
    113 000136                       NTLER$  <:>,15,NONE,$NIERR,,,<NS: Assignment Failure>
    114 000176                       NTLER$  <:>,17,NONE,$NIERR,,,<Network ACP (NETACP) Not Installed>
    115 000252                       NTLER$  ,18,NONE,$NIERR,,,<PLB Allocation Failure>
    116 000312                       NTLER$  <:>,19,NONE,$NIERR,,,<Routing Control Task Not Installed>
    117                                     .DSABL  LC
    118
    119                              ;
    120                              ; START OF PURE CODE
    121                              ;
    122 000000                              .PSECT
```

```
NTILON11S  CREATED BY  MACRO  ON 29-JUN-85 AT 03:31      PAGE 4
MACRO CROSS REFERENCE                                    CREF    04.00

MACRO NAME       REFERENCES

ADJDF$           #6-62        6-73
ALUN$            #6-61        7-105
BIAS             #6-61
CACDF$           #6-62        6-74
CALL              9-141       9-160        9-178        9-185        9-253        9-254        9-276
CALLX            #6-62
CCBDF$           #6-63        6-65
CTRDF$           #6-61        6-71
DCBDF$           #6-63        6-77
DHBDF$           #6-62        6-75
DIR$             #6-60        9-150        9-265       #9-274        9-274
DVBDF$           #6-63        6-76
ECDDB$           #6-61        6-69
EMSG$            #6-60
EMSG$R           #6-60        9-152        9-192        9-195        9-294        9-295        9-296        9-297
HDRDF$           #6-63        6-79
MAP              #6-61
NHWDF$           #6-60        6-66
NSFDF$           #6-62        6-72
NTLER$           #6-60        8-112        8-113        8-114        8-115        8-116
OFF$             #7-104      #7-105
PDVDF$           #6-61        6-70
PLBDF$           #6-61        6-67
RESMAP           #6-62
RETC             #6-60
RETURN            9-148       9-188        9-260        9-278
RUN$             #6-60        7-104
R50$             #7-104       7-104
SAVMAP           #6-62
SOB               9-259
SWSTK$            9-160       9-253
UCBDF$           #6-63        6-78
WSIG$S           #6-60        9-274
XPDDB$           #6-61        6-68
```

FILEID**NTIMEF

L 7

```
NTIMFE - DO MARKED-FOR-ENABLE S MACRO V05.03b Wednesday 24-Jul-85 15:52  Page 9-4
Symbol table

ZF.INI= 040000          ZF.MUX= 000040          Z.AVL    000014          Z.MAP    000020          $HEADR= ****** GX
ZF.KMX= 000020          ZF.PSE= 002000          Z.DAT    000016          Z.NAM    000004          $NIMFE  000006RG
ZF.LLC= 000004          ZF.SLI= 010000          Z.DSP    000000          Z.PCB    000012          $SAVAL= ****** GX
ZF.LMC= 000100          ZF.TIM= 000200          Z.FLG    000010          Z.SCH    000007          $$$ARG= 000004
ZF.MAN= 020000          ZF.X3P= 000000          Z.LEN =  000016          $DBECL= ****** GX         $$$R  = 000010
ZF.MFL= 000010          ZS.ASN= 100000          Z.LLN    000006          $ERR12= ****** GX         .$$$$.= 000034
ZF.MTM= 000400          ZS.BSY= 140000

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000756    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 113
Work  file writes: 106
Size of work file: 27005 Words  ( 106 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:41.24
DB2:NTIMFE11S.T34,[132,134]NTIMFE11S/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]T34,NTIMFE
```

M 7

```
NTIMFE11S  CREATED BY  MACRO  ON 24-JUL-85 AT 15:52      PAGE 1
```

```
    213
    214 000432  000000G                 UMRPT:: $UMRPT
    215 000434  000000G                 SRSTD:: $SRSTD
    216 000436  000000G                 ALOCB:: $ALOCB
    217 000440  000000G                 DEACB:: $DEACB
    218 000442  000000G                 PUMR::  $PUMR
    219 000444  000000G                 HEADR:: $HEADR
    220 000446  000000G                 TKTCB:: $TKTCB
    221 000450  000000G                 QASTT:: $QASTT
    222 000452  000000G                 CLKHD:: $CLKHD
    223 000454  000000G                 TKPS::  $TKPS
    224 000456  000000G                 ABTIM:: $ABTIM
    225 000460  000000G                 DEVHD:: $DEVHD
    226 000462  000000G                 LDPWF:: $LDPWF
    227
    228                                         .IF DF  R$$MPL
    229                                 MPPRO:: $MPPRO
    230                                         .ENDC
    231
    232 000464  000000G                 SLTMA:: $SLTMA                  ; COMM EXEC VECTORED VALUES
    233 000466  000000G                 SLTNM:: $SLTNM
    234 000470  000000G                 NTLPT:: $NTLPT
    235 000472  000000G                 CALLX:: $CALLX
    236 000474  000000G                 T100C:: $T100C
    237 000476  000000G                 T100Q:: $T100Q
    238 000500  000000G                 TK100:: $TK100
    239 000502  000000G                 CCBGT:: $CCBGT
    240 000504  000000G                 LLCRS:: $LLCRS
    241 000506  000000G                 PSIPT:: $PSIPT
    242 000510  000000G                 CEACC:: $CEACC
    243 000512  000000G                 T1SCL:: $T1SCL
    244 000514  000000G                 LGSTT:: $LGSTT
    245 000516  000000G                 DECPT:: $DECPT
    246 000520  000000G                 PDVNM:: $PDVNM
    247 000522  000000G                 PDVID:: $PDVID
    248 000524  000000G                 LLCAL:: $LLCAL
    249 000526  000000G                 LLCTA:: $LLCTA
    250 000530  000000G                 PDVTA:: $PDVTA
    251
    252         000043                          VECLN == <<.-EXEVEC>/2>
    253
    254                                 ;
    255                                 ;   START OF PURE CODE
    256                                 ;
    257 000000                                  .PSECT
```

```
                        .SBTTL  RCV DATA AST

                ;+
                ; NTIRDA - NTINIT RCV DATA AST
                ;
                ; INPUTS:
                ;       NONE
                ;
                ; OUTPUTS:
                ;       THE RCV DATA SUBROUTINE IS CALLED UNTIL THE RCV DATA QUEUE
                ;         IS EMPTIED.
                ;
                ;-

                        .IF NDF R$$11S

                NTIRDA: MOV     R0,-(SP)                ;SAVE R0
                10$:    DIR$    #NTIRCV                 ;RECEIVE DATA
                        BCS     20$                     ; .. NONE
                        CALL    NTIRDT                  ;PROCESS THE DATA
                        BR      10$                     ;TRY FOR SOME MORE
                20$:    MOV     (SP)+,R0                ;RESTORE R0
                        ASTX$S                          ;EXIT FROM AST

                        .ENDC
```

```
                                                                    L 10
NTINIT11S  CREATED BY  MACRO  ON 5-SEP-85 AT 11:10      PAGE 7
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME      REFERENCES

ASTX$S          #6-59
CALL             10-378   10-380    10-387    10-404    13-590    13-592    13-602    14-681    15-716    15-726
                 15-737
CALLR           #6-61     15-730    15-742
CALLX           #6-61
CCBDF$          #6-61     6-73
CLEF$S          #6-59    #6-60
DIR$            #6-60     9-309     10-401   #10-402    10-402    10-403   #10-463    10-463   #14-689    14-689
DSAR$S          #6-60
EMSG$           #6-59     10-378    10-380
EMSG$R          #6-62
ENAR$S          #6-60
EXIT$S          #6-59     10-463
MAP             #6-61
MOV$            #10-402   10-402   #14-689    14-689    14-689    14-689   #14-689    14-689    14-689   #14-689
                 14-689   14-689
MRKT$           #6-62     7-109
MVB$            #14-689   14-689
NHWDF$          #6-61     6-77
NTLDF$          #6-61     6-72
NTLER$          #6-59     8-192     8-193     8-194
OFF$            #7-104   #7-109    #7-110
OPTDF$          #6-61     6-74
PCBDF$          #6-63     6-78
PDVDF$          #6-61     6-75
QDPB$S          #14-689   14-689
QIOW$S          #6-59     14-689
RCVD$           #6-60
RCVX$           #6-62     7-104
RESMAP          #6-63
RETC            #6-62
RETURN           13-611   14-697    15-720    15-743
RQST$           #6-60     7-110
RVP$            #14-689   14-689    14-689    14-689
R50$            #7-104    7-104    #7-110     7-110
SAVMAP          #6-63
SCBDF$          #6-63     6-79
SDAT$           #6-60
SDAT$S          #6-62
SETF$S          #6-60
SLTDF$          #6-61     6-76
SRDA$S          #6-60
SWSTK$           13-590
TCBDF$          #6-63     6-80
VECDF$          #6-62     6-81
WSIG$S          #6-59
WTSE$S          #6-59     10-402
```

M 10

```
NTIUMR  MACRO V05.03b Saturday 29-Jun-85 03:33
Table of contents

     6-   54     DATA
     7-   69     LOAD UMR'S
```

```
NTIUMR  MACRO V05.03b Saturday 29-Jun-85 03:33  Page 5
```

```
                                         .SBTTL  $CFCOM - GET A COMMAND LINE
                                 ;+
                                 ; *** - $CFCOM - GET A COMMAND LINE
                                 ;
                                 ; INPUTS:
                                 ;         NONE.
                                 ;
                                 ; OUTPUTS:
                                 ;         C-BIT = SUCCESS/FAILURE INDICATOR
                                 ;         ALL REGISTERS = DESTROYED
                                 ;
                                 ;-

000510                           $CFCOM::
000510  005067  000000G          10$:    CLR     $CMAND          ; INITIALIZE SYMBOLS
000514  005067  000000G                  CLR     $OPRND          ;       ...
000520  005067  000000G                  CLR     $OPRN2          ;       ...
000524  005067  000000G                  CLR     $COMP           ;       ...
000530  005067  000000G                  CLR     $SWICH          ;       ...
000534  005067  000000G                  CLR     $FLAGS          ;       ...
000540  005067  000000G                  CLR     $WILD           ;       ...
000544  005067  000000G                  CLR     $CIRCT          ;       ...
000550  005067  000000G                  CLR     $INFO           ;       ...
000554  012767  177777  000000G          MOV     #-1,$TNUM       ; ASSUME ''KNOWN OBJECTS''
000562  005067  000000G                  CLR     $NODNA          ; ASSUME ''KNOWN REMOTES''
000566  005067  000002G                  CLR     $NODNA+2        ;       ...
000572  005067  000000G                  CLR     $VLVL           ; ZERO THE OBJECT TASK VERIFICATION LEVEL

000576  012701  001400                   MOV     #<3*400>,R1     ; 3 CHARS MUST MATCH
000602  012702  000000G                  MOV     #PRSKTB,R2      ; POINT R2 AT KEYWORD TABLE
000606  012705  000000G                  MOV     #PRSSTB,R5      ; POINT R5 AT STATE TABLE
000612  012767  000000G 177666           MOV     #ERRILC,$ERROR  ; ASSUME ILLEGAL COMMAND
000620  004767  000000G                  CALL    $GTPLN          ; PARSE THE COMMAND LINE
000624  103054                           BCC     30$             ; IF CC, ALL OKAY
000626  102441                           BVS     20$             ; IF VS, GCL ERROR
                                 ;
                                 ; TPARS ERROR
                                 ;
000630                                   EPRINT  #CRLF           ; TERMINATE LINE AND INDENT
000640  111446                           MOVB    (R4),-(SP)      ; SAVE BYTE AFTER LAST
000642  112714  000200                   MOVB    #200,(R4)       ; TERMINATE LINE
000646                                   EPRINT  #$BUFF          ; PRINT GOOD PART OF LINE
000656  112614                           MOVB    (SP)+,(R4)      ; RESTORE REMAINDER
000660                                   EPRINT  #ARROWS         ; ARROWS...
000670  010400                           MOV     R4,R0           ; COPY POINTER
000672  105724                   15$:    TSTB    (R4)+           ; LOOK FOR NULL
000674  001376                           BNE     15$             ; IF NE, CONTINUE
000676  112744  000200                   MOVB    #200,-(R4)      ; SUPPRESS CARRAIGE CONTROL
000702                                   EPRINT  R0              ; PRINT BAD PART OF LINE
000706  105014                           CLRB    (R4)            ; RESTORE NUL BYTE
000710                                   EPRINT  #CR             ; END LINE
000720                                   ERROR$  $ERROR          ; PRINT ERROR MESSAGE
000730  000667                           BR      10$             ; GET NEXT LINE
                                 ;
                                 ; CHECK GCL ERROR
                                 ;
000732  122777  000000G 000000G  20$:    CMPB    #IE.EOF,@$CLIOS ; JUST AN END-OF-FILE ?
```

```
000424  001421                           BEQ     60$                ; IF EQ, GO UPDATE LOW SUBADDRESS
000426  005716                           TST     (SP)               ; ANYTHING PRESENT ?
000430  001417                           BEQ     60$                ; IF EQ, NO - SET UP LOW
000432  020160  000000G                  CMP     R1,P$STLO(R0)      ; ELSE LOOK FOR BEST FIT
000436  101414                           BLOS    60$                ; IF LOS, CLOSEST TO LOW
000440  010102                   40$:    MOV     R1,R2              ; UPDATE HIGH SUBADDRESS
000442  000417                           BR      70$                ; ...
                                 ;
                                 ; USER SPECIFIED RANGE ON COMMAND LINE
                                 ;
000444  042760  000000C 000000G  50$:    BIC     #CS.THI!CS.TLO,C.STS(R0) ; ASSUME CLEARING SUBADDRESSES
000452  020102                           CMP     R1,R2              ; IN CORRECT ORDER ?
000454  103405                           BLO     60$                ; IF LO, YES
000456  010146                           MOV     R1,-(SP)           ; ELSE PUT THEM
000460  010201                           MOV     R2,R1              ; IN ORDER
000462  011602                           MOV     (SP),R2            ; ...
000464  060126                           ADD     R1,(SP)+           ; WERE THEY BOTH ZERO ?
000466  001414                           BEQ     80$                ; IF EQ, YES - ALL CLEARED
                                 ;
                                 ; SET NEW SUBADDRESS(ES)
                                 ;
000470  010160  000000G          60$:    MOV     R1,P$STLO(R0)      ; STUFF LOW SUBADDRESS
000474  052760  000000G 000000G          BIS     #CS.TLO,C.STS(R0)  ; REMEMBER IT
000502  005702                   70$:    TST     R2                 ; GOT PART 2 ?
000504  100405                           BMI     80$                ; IF MI, WASN'T ONE
000506  010260  000000G                  MOV     R2,P$STHI(R0)      ; STUFF HIGH SUBADDRESS
000512  052760  000000G 000000G          BIS     #CS.THI,C.STS(R0)  ; ELSE REMEMBER IT

000520  005726                   80$:    TST     (SP)+              ; CLEAN EL STACKO

000522  000207                   90$:    RETURN
                                 ;
                                 ; ERRORS
                                 ;
000524                           100$:   ERROR$  #ERRPSI            ; PSI ONLY PARAMETER
000534  000207                           RETURN
```

Table of contents

5-   69 $CFDSP - DISPATCH ON THE COMMAND CODE

```
CFEHLP     CREATED BY  MACRO  ON 29-JUN-85 AT 04:55      PAGE 2
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME      REFERENCES

CALLR           #4-79       5-117
DIR$            #4-79
ERROR$          #4-64
FDBDF$          #4-88       4-88
FDBSZ$          #4-88       4-88
HLPDF$          #4-79       4-88
PRINT           #4-59
```

```
CFEIMP - SYMBOL DEFINITIONS AND MACRO V05.03b Saturday 29-Jun-85 04:55  Page 9-1
DEFINE TEMPLATES GLOBALLY

    401 000016                                    .ASECT
    402           000000                          .=0
    403 000000                                    .BLKW    1               ; C.LNK
    404 000002                                    .BLKW    1               ; C.STS
    405 000004                           U$NUNT::.BLKW    1               ; UNIT NUMBER
    406 000006                           U$NCW0::.BLKW    1               ; CHARACTERISTICS WORD 0
    407 000010                           U$NCW1::.BLKW    1               ; CHARACTERISTICS WORD 1
    408 000012                           U$NCSR::.BLKW    1               ; SECONDARY CSR ADDRESS
    409 000014                           U$NCST::.BLKW    1               ; LINE COST
    410 000016                           U$NDPR::.BLKW    1               ; DEAD POLLING RATIO
    411 000020                                    .BLKW    1               ; UNUSED
    412 000022                                    .BLKW    1               ; UNUSED
    413 000024                           U$NPCH::.BLKW    1               ; EMULATOR CHARACTERISTICS WORD
    414           000026                 U$NLEN==.
    415 002054                                    .PSECT
    416
    417                                  ;*****
    418                                  ;
    419                                  ; DEFINE STATION TEMPLATE
    420                                  ;
    421 000026                                    .ASECT
    422           000000                          .=0
    423 000000                                    .BLKW    1               ; S$TLNK
    424 000002                                    .BLKW    1               ; S$TSTS
    425 000004                           S$TSTA::.BLKB    1               ; STATION NUMBER
    426 000005                           S$TADD::.BLKB    1               ; STATION ADDRESS
    427 000006                           S$TFLG::.BLKB    1               ; STATION FLAGS
    428 000007                           S$TCST::.BLKB    1               ; STATION COST
    429 000010                           S$TAPR::.BLKB    1               ; ACTIVE POLLING RATIO
    430 000011                                    .BLKB    1               ; AVAILABLE
    431 000012                           S$THTM::.BLKW    1               ; HELLO TIMER
    432 000014                                    .BLKW    1               ; UNUSED
    433           000016                 S$TLEN==.
    434 002054                                    .PSECT
    435
    436                                  ;****
    437                                  ;
    438                                  ; DEFINE LLC TEMPLATE
    439                                  ;
    440 000016                                    .ASECT
    441           000000                          .=0
    442 000000                                    .BLKW    1               ; C.LNK
    443 000002                                    .BLKW    1               ; C.STS
    444 000004                           L$LNAM::.BLKW    1               ; LLC PROCESS NAME
    445 000006                           L$LFLG::.BLKW    1               ; LLC FLAGS WORD
    446 000010                           L$LPRI::.BLKB    1               ; LLC PROCESS PRIORITY
    447 000011                           L$LLIN::.BLKB    1               ; NUMBER OF LINES ASSIGNED TO THIS LLC
    448 000012                           L$LDEV::.BLKW    1               ; PSEUDO DEVICE NAME FOR THIS LLC
    449 000014                           L$LNLT::.BLKB    1               ; NUMBER OF LINE TABLE TO ALLOCATE
    450 000015                           L$LEXT::.BLKB    1               ; PROCESS EXTENSION IN BLOCKS
    451 000016                           L$LCTM::.BLKW    1               ; COUNTER TIMER
    452           000020                 L$LLEN==.
    453 002054                                    .PSECT
    454
    455                                  ;****
    456                                  ;
    457                                  ; DEFINE THE PSN BLOCK
```

```
          .TITLE  NTIX25 - NTINIT 'MODULE X25-SERVER ON' ROUTINE
          .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;         NTINIT - ''SET MODULE X25-SERVER'' PROCESSING
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00    11-MAR-81
;         VERSION 3.1 RELEASE
;
; 3.00    16-APR-82
;         DECNET-11M V3.1
;         DECNET-11M-PLUS V1.1
;
; 4.00    07-NOV-83
;         DECNET-11M V4.0
;         DECNET-11M-PLUS V2.0
;
; 5.00    22-JUL-85
;         DECnet-11M/S V4.2
;         DECnet-11M-Plus V3.0
;         DECnet-Micro/RSX V1.0
;
;

```

```
                        .TITLE  NTICEX
                        .IDENT  /V05.00/
                ;
                ; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                ;
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                ; TRANSFERRED.
                ;
                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                ; CORPORATION.
                ;
                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                ;
                ; MODULE DESCRIPTION:
                ;
                ;       NTINIT - "SET CEX" PROCESSING
                ;
                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                ;
                ; IDENT HISTORY:
                ;
                ; 1.00  27-FEB-78
                ;       VERSION 2.0 RELEASE
                ;
                ; 2.00  14-DEC-79
                ;       DECNET-11M/S V3.0
                ;       DECNET-11M-PLUS V1.0
                ;
                ; 3.00  16-APR-82
                ;       DECNET-11M V3.1
                ;       DECNET-11M-PLUS V1.1
                ;
                ; 4.00  07-NOV-83
                ;       DECNET-11M V4.0
                ;       DECNET-11M-PLUS V2.0
                ;
                ; 5.00  22-JUL-85
                ;       DECnet-11M/S V4.2
                ;       DECnet-11M-Plus V3.0
                ;       DECnet-Micro/RSX V1.0
                ;
                ;
```

```
**FILE**ID**NTICSR
```

```
NN      NN    TTTTTTTTTT      IIIIII        CCCCCCCC      SSSSSSSS    RRRRRRRR
NN      NN    TTTTTTTTTT      IIIIII        CCCCCCCC      SSSSSSSS    RRRRRRRR
NN      NN        TT            II        CC            SS            RR      RR
NN      NN        TT            II        CC            SS            RR      RR
NNNN    NN        TT            II        CC            SS            RR      RR
NNNN    NN        TT            II        CC            SS            RR      RR
NN  NN  NN        TT            II        CC              SSSSSS      RRRRRRRR
NN  NN  NN        TT            II        CC              SSSSSS      RRRRRRRR
NN    NNNN        TT            II        CC                    SS    RR  RR
NN    NNNN        TT            II        CC                    SS    RR  RR
NN      NN        TT            II        CC                    SS    RR    RR        ....
NN      NN        TT            II        CC                    SS    RR    RR        ....
NN      NN        TT          IIIIII        CCCCCCCC    SSSSSSSS      RR      RR      ....
NN      NN        TT          IIIIII        CCCCCCCC    SSSSSSSS      RR      RR      ....
```

```
     11           11           SSSSSSSS
     11           11           SSSSSSSS
   1111         1111         SS
   1111         1111         SS
     11           11         SS
     11           11         SS
     11           11           SSSSSS
     11           11           SSSSSS
     11           11                 SS
     11           11                 SS
     11           11                 SS
     11           11                 SS
   111111       111111       SSSSSSSS
   111111       111111       SSSSSSSS
```

NTICSR MACRO V05.03b Saturday 29-Jun-85 03:30

```
        .TITLE  NTIDLI - NTINIT DOWN LINE LOAD
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       NTINIT - COPY DOWN-LINE SYSTEM LOAD INFORMATION INTO LOW CORE
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;


```

```
                                        .SBTTL  SET EXECUTOR ON

                                ;+
                                ; $NILON - SET STATE EXECUTOR ON
                                ;
                                ; REQUEST THE NETACP TO START
                                ;
                                ; INPUTS:
                                ;         NETACP INSTALLED
                                ;
                                ; OUTPUTS:
                                ;         THE DEVICE LIST IS SCANNED TO FIND THE NETACP.  IF IT IS FOUND,
                                ;         A VCB IS ALLOCATED FOR IT, AND IT IS REQUESTED TO RUN.
                                ;         $NIRDB IS SET TO INDICATE SUCCESS (+) OR FAILURE (-) OR NO-OP (0).
                                ;         CARRY IS SET ON FAILURE.
                                ;-
000000                          $NILON::
000000                                  CALL    FNDPDV          ;FIND NSP PROCESSES (ECL & XPT)
000004  005767  000000G                 TST     $DBXPT          ;BOTH TRANSPORT (XPT)
000010  001403                          BEQ     10$             ;  AND ECL MUST BE
000012  005767  000000G                 TST     $DBECL          ;    PRESENT IN THE SYSTEM
000016  001002                          BNE     20$             ;      AND BE LOADED

000020  000261                  10$:    SEC
000022                                  RETURN

000024                          20$:    DIR$    #ASSIGN         ;ASSIGN LUN 2 TO NS:
000032  103003                          BCC     30$             ; IF CS, ERROR
000034                                  EMSG$R  15              ;ALUN$ FAILURE OR NS: LOAD FAILURE

000042  016700  000000G         30$:    MOV     $NIXPT,R0       ;GET PDV INDEX OF XPT
000046  067700  000000G                 ADD     @PDVTA,R0       ;POINT INTO PDV INDEX TABLE
000052  011000                          MOV     (R0),R0         ;GET ADDRESS OF XPT'S PDV
000054  017701  000000G                 MOV     @DECPT,R1       ;GET DECNET HOME BLOCK ADDRESS
000060  116061  000006  000030          MOVB    Z.LLN(R0),D$NLN(R1)     ;INITIALIZE THE NUMBER OF ROUTIN

000066                                  SWSTK$  60$             ;* ENTER SYSTEM STATE
000072  005066  000004                  CLR     4(SP)           ;* CLEAR USER R1

                                        .IF     NDF,R$$MPL
000076  017703  000000G                 MOV     @HEADR,R3       ;* GET TASK HEADER
000102  016303  000102                  MOV     H.LUN+4(R3),R3  ;* ADDRESS OF NS: UCB
                                        .IFF
                                        SAVMAP                  ;* SAVE APR6 CONTENTS
                                        MAP     @SAHDB          ;* MAP TO HEADER IN SECONDARY PO
                                        MOV     @SAHPT,R3       ;* GET THE HEADER POINTER
                                        MOV     H.LUN+4(R3),R3  ;* ADDRESS OF NS: UCB
                                        RESMAP                  ;* RESTORE APR6
                                        .ENDC

000106  016301  000014                  MOV     U.CW3(R3),R1    ;* GET FEATURES WORD FOR THIS NODE
000112  032701  000200                  BIT     #NF.END,R1      ;* IS THIS A FULL ROUTING NODE?
000116  001020                          BNE     50$             ;* IF NE, NO
000120  012703  000000'                 MOV     #RCP1NM,R3      ;* POINT TO TASK NAME
000124                                  CALL    @SRSTD          ;* IS ROUTING CONTROL PROGRAM IN
000130  103407                          BCS     40$             ;* IF CS, NOT INSTALLED
```

```
**FILE**ID**NTIMFE





NN      NN  TTTTTTTTTT    IIIIII    MM      MM  FFFFFFFFFF  EEEEEEEEEE
NN      NN  TTTTTTTTTT    IIIIII    MM      MM  FFFFFFFFFF  EEEEEEEEEE
NN      NN      TT          II      MMMM  MMMM  FF          EE
NN      NN      TT          II      MMMM  MMMM  FF          EE
NNNN    NN      TT          II      MM  MM  MM  FF          EE
NNNN    NN      TT          II      MM  MM  MM  FF          EE
NN  NN  NN      TT          II      MM      MM  FFFFFFFF    EEEEEEEE
NN  NN  NN      TT          II      MM      MM  FFFFFFFF    EEEEEEEE
NN    NNNN      TT          II      MM      MM  FF          EE
NN    NNNN      TT          II      MM      MM  FF          EE
NN      NN      TT          II      MM      MM  FF          EE              ....
NN      NN      TT          II      MM      MM  FF          EE              ....
NN      NN      TT        IIIIII    MM      MM  FF          EEEEEEEEEE      ....
NN      NN      TT        IIIIII    MM      MM  FF          EEEEEEEEEE      ....




    11          11        SSSSSSSS
    11          11        SSSSSSSS
  1111        1111      SS
  1111        1111      SS
    11          11      SS
    11          11      SS
    11          11        SSSSSS
    11          11        SSSSSS
    11          11              SS
    11          11              SS
    11          11              SS
    11          11              SS
  111111      111111    SSSSSSSS
  111111      111111    SSSSSSSS
```

```
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE          REFERENCES

ACTCIR    000004 R     #6-93     *7-119     7-190     *7-196     7-234     *7-240
CCBGT   = ******  GX    8-289
CTLNUM    000003 R     #6-92     *7-166
C.BUF1    000014       *8-291
C.FNC     000010       *8-296
C.LIN     000006       *8-293
C.NSP     000004       *8-294
C.STA     000007       *8-292
C.STS     000012       *8-295
DECPT   = ******  GX    7-187     7-231
D$NN      000060        7-188     7-232
ENAC      000074 R     #7-123     7-249
ENALIN    000610 R      7-198     7-241     #8-269
E$NBR     000014       #6-72
E$NBS     000020       #6-72
E$NCR     000034       #6-72
E$NCS     000036       #6-72
E$NIC     000044       #6-72
E$NLEN    000050       #6-72
E$NLLA    000012       #6-72
E$NLNK    000000       #6-72
E$NML     000040       #6-72
E$NMR     000024       #6-72
E$NMS     000030       #6-72
E$NNOD    000002       #6-72
E$NRT     000042       #6-72
E$NRTP    000005       #6-72
E$NSEG    000010       #6-72
E$NTIM    000046       #6-72
E$NUSE    000004       #6-72
E$STRT    000006       #6-72
FC.CTL  = 000006        8-296
FNDPDV  = ******  GX    7-118
FS.STR  = 001000        8-296
I$$AS   = ******        6-77
LF.ACT  = 100000       #6-74      7-132     7-200     7-243
LF.BRO  = 000400       #6-74      7-135
LF.BWT  = 000007       #6-74
LF.ENA  = 002000       #6-74      7-214     7-236
LF.LPB  = 001000       #6-74
LF.MDC  = 000100       #6-74
LF.MFL  = 004000       #6-74
LF.MTP  = 000020       #6-74
LF.PAC  = 000200       #6-74
LF.RDY  = 040000       #6-74      7-125
LF.REA  = 010000       #6-74
LF.SER  = 000040       #6-74
LF.TIM  = 000010       #6-74
LF.UNL  = 020000       #6-74
LF.X2P  = 000000       #6-74
LLCRS   = ******  GX    8-297
LLCTA   = ******  GX    7-121
```

```
                                .SBTTL NETWORK INITIALIZATION COLD START

                        ;+
                        ; NTINIT - ENTER THE NTINIT TASK
                        ;
                        ; THIS ENTRY SHOULD ONLY BE TAKEN ONCE, AS IT WILL STAY ACTIVE,
                        ; WAITING FOR MORE INSTRUCTIONS VIA ''SEND DATA''.
                        ;
                        ; INPUTS:
                        ;       NTINIT GET INPUT VIA ''RCV DATA'' AND RETURNS COMPLETION VIA
                        ;         ''SEND DATA''
                        ;       THE FIRST WORD OF THE RCV DATA PACKET ON RCV INDICATES THE
                        ;         FUNCTION TO BE PERFORMED
                        ;       THE FIRST WORD OF THE SEND DATA PACKET ON COMPLETION INDICATES
                        ;         SUCCESS (+), FAILURE (-)  OR NO-OP (0)
                        ;       WHEN SENDING A FUNCTION REQUEST TO NTINIT, THE REQUEST SHOULD
                        ;         BE QUEUED FIRST (VIA SEND DATA TO NTINIT) AND NTINIT SHOULD
                        ;         THEN BE REQUESTED
                        ;       EVENT FLAGS START OFF ''CLEAR''
                        ;
                        ; OUTPUTS:
                        ;       CHECK COMPATIBILITY OF RSX SYSTEM
                        ;       START MAIN CONTROL LOOP
                        ;-

000000                  NTINIT:
                                .IF DF  R$$MPL
                                .IF NDF R$$PRO
                                DIR$    #TRNVEC                 ; INITIALIZE VECTOR TABLE
                                BCC     1$                      ; IF SUCCESS - BRANCH
                                IOT                             ; THIS SOULD NEVER OCCUR
                        1$:     BIT     #FE.CEX,@FMASK          ; COMM EXEC LOADED ?
                                BEQ     5$                      ; IF NOT - BRANCH
                                CALL    VECSET                  ; IF YES - INITIALIZE PROCESS VECTOR TABLES
                        5$:
                                .ENDC
                                .ENDC

                                .IF NDF R$$11S
                                CLEF$S  #15.                    ;CLEAR EVENT FLAG
                                CALL    NTIRDX                  ;GET FIRST RECEIVE DATA OR EXIT
                                SRDA$S  #NTIRDA                 ;SET THE RCV DATA AST ENTRY

                                CLR     $NIAST                  ;ASSUME CALLER NOT NOTIFIED VIA AST
                                BIT     #LS.AST,$NIFNC          ;NOTIFY CALLER VIA AST?
                                BEQ     10$                     ;BR IF NO
                                MOV     LR.AST+$NIRDB,$NIAST    ;STORE ADDRESS OF NOTIFICATION AST
                                MOV     LR.PRM+$NIRDB,$NIPRM    ;STORE NOTIFICATION AST PARAMETER
                        10$:
                                .IFF
000000                          DIR$    #NTIRCX                 ;RECEIVE DATA OR EXIT
                                .ENDC

                                .IF DF S$$BAS
                                                                ; FOR THE SERVER BASE, WE NEED TO
                                                                ; DO SOME OF THE THINGS WE WOULD
                                                                ; DO FOR 11M - THIS IS CODE COPIED FROM
```

```
        .SBTTL  RCV DATA SUBROUTINE

;+
; NTIRDX - RCV DATA AND RETURN OR EXIT
; NTIRDT - PROCESS THE RECEIVED DATA
;
; THESE ROUTINES GET A RCV DATA PACKET FROM EITHER A FUNCTION REQUESTOR
; OR FROM NTL.
;
; INPUTS:
;       LOW BYTE OF $NIFLG IS EITHER ZERO (RCV FUNCTION REQUEST) OR
;         NON-ZERO (BUSY WITH FUNCTION REQUEST, RCV ONLY FROM NTL)
;
; OUTPUTS:
;       THE DATA IS RECEIVED INTO "$NIRTN,$NIRDB".
;       IF THE LOW BYTE OF $NIFLG IS ZERO, THE SENDERS NAME IS SAVED
;         IN THE "NTISEN" DIRECTIVE, THE FUNCTION CODE IS SAVED
;         IN "$NIFNC".
;       IF THE LOW BYTE OF $NIFLG IS NON-ZERO, AND THE SENDER IS NTL,
;         NTL'S ANSWER IS SAVED IN "$NTLFC" AND EVENT FLAG 15. IS
;         SET, ELSE A "BUSY" RESPONSE IS SENT TO THE REQUESTOR.
;
;-
        .IF NDF R$$11S

NTIRDX: DIR$    #NTIRCX                 ;RECEIVE DATA OR EXIT

NTIRDT: MOV     #$NIRTN,R0              ;POINT AT SENDING TASK'S NAME
        TSTB    $NIFLG                  ;ARE WE ALREADY ACTIVE ?
        BEQ     30$                     ;BR IF NO, RCV FUNCTION REQUEST

        CMP     #^RNTL,(R0)+            ;IS DATA FROM NTL ?
        BNE     20$                     ;BR IF NO, ERROR
        CMP     #^R...,(R0)+            ;MAYBE ?
        BNE     10$                     ;BR IF NO, STILL AN ERROR
        MOV     (R0),$NTLFC             ;SAVE NTL'S ANSWER
        SETF$S  #15.                    ;SET THE EVENT FLAG
        BR      40$                     ;AND LEAVE

10$:    TST     -(R0)                   ;RESET THE POINTER
20$:    MOV     (R0),NTIBSY+S.DATN+2    ;SET THE REQUESTORS TASK NAME INTO
        MOV     -(R0),NTIBSY+S.DATN     ;  THE BUSY RESPONSE DIRECTIVE
        DIR$    #NTIBSY                 ;TELL REQUESTOR WE ARE BUSY
        BR      40$                     ;EXIT WITH CARRY CLEAR

30$:                                    ; (NOTE THE FOLLOWING 5 LINES ARE
                                        ; DUPLICATED IN SERVER BASE CODE
                                        ; AFTER NTINIT - PLEASE MAKE CHANGES
                                        ; TO BOTH PLACES)
        MOV     (R0)+,NTISEN+S.DATN     ;SAVE SENDER'S NAME IN
        MOV     (R0)+,NTISEN+S.DATN+2   ;  RETURN DIRECTIVE
        MOV     (R0)+,$NIFNC            ;SAVE FUNCTION CODE
        MOV     (R0)+,$NIFNC+2          ; SAVE TWO WORDS OF SENDER'S DATA
        MOV     (R0),$NIFNC+4           ; (USED FOR POWERFAIL RECOVERY)
        INCB    $NIFLG                  ;WE ARE NOW BUSY

40$:    RETURN
```

```
**FILE**ID**NTITIM





NN        NN  TTTTTTTTTT    IIIIII    TTTTTTTTTT    IIIIII    MM        MM
NN        NN  TTTTTTTTTT    IIIIII    TTTTTTTTTT    IIIIII    MM        MM
NN        NN      TT          II          TT          II      MMMM    MMMM
NN        NN      TT          II          TT          II      MMMM    MMMM
NNNN      NN      TT          II          TT          II      MM  MM  MM
NNNN      NN      TT          II          TT          II      MM  MM  MM
NN  NN    NN      TT          II          TT          II      MM        MM
NN  NN    NN      TT          II          TT          II      MM        MM
NN    NNNN        TT          II          TT          II      MM        MM
NN    NNNN        TT          II          TT          II      MM        MM
NN        NN      TT          II          TT          II      MM        MM        ....
NN        NN      TT          II          TT          II      MM        MM        ....
NN        NN      TT        IIIIII        TT        IIIIII    MM        MM        ....
NN        NN      TT        IIIIII        TT        IIIIII    MM        MM        ....




    11          11        SSSSSSSS
    11          11        SSSSSSSS
  1111        1111      SS
  1111        1111      SS
    11          11      SS
    11          11      SS
    11          11        SSSSSS
    11          11        SSSSSS
    11          11              SS
    11          11              SS
    11          11              SS
    11          11              SS
  111111      111111    SSSSSSSS
  111111      111111    SSSSSSSS
```

```
          .TITLE  NTIUMR
          .IDENT  /V05.00/

;
; COPYRIGHT (C) 1978,1979,1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
; MODULE DESCRIPTION:
;
;       NTINIT - LOAD UNIBUS MAPPING REGISTERS FOR NETWORK BUFFER POOL
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  27-FEB-78
;       VERSION 2.0 RELEASE
;
; 2.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  7-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
CFECOM - GET A COMMAND LINE     MACRO V05.03b Saturday 29-Jun-85 04:54  Page 5-1
$CFCOM - GET A COMMAND LINE

    166 000740  000261                          SEC                     ; ASSUME SO
    167 000742  001405                          BEQ     30$             ; IF EQ, YES
    168 000744                                  ERROR$  #ERRGCL         ; ELSE, COMMAND LINE ERROR
    169 000754  000655                          BR      10$             ; AND GET NEXT LINE
    170 000756  000207                  30$:    RETURN
    233                                                      .SBTTL  SETLIN - SET LINE PARAMETERS
    234                                             ;+
    235                                             ; *** - SETLIN - SET LINE PARAMETERS
    236                                             ;
    237                                             ; INPUT:
    238                                             ;       $OPRND = OPERANDS PRESENT
    239                                             ;
    240                                             ; OUTPUT:
    241                                             ;       CONTROL PASSED TO PROPER ROUTINE
    242                                             ;
    243                                             ;-
    244
    245 000536                                      SETLIN:
    246 000536  032767  000000G 000000G                     BIT     #OL.SLT,$OPRND  ; CHANGE THE SLT/STATION DEFINITION ?
    247 000544  001402                                      BEQ     10$             ; IF EQ, NO
    248 000546  004767  000000G                             CALL    $SESLT          ; CHANGE THE SLT DEFINITION
    249 000552  032767  000000G 000000G 10$:                BIT     #OL.CNT,$OPRND  ; CHANGE THE CONTROLLER DEFINITION ?
    250 000560  001403                                      BEQ     20$             ; IF EQ, NO
    251 000562  004767  000000G                             CALL    $SECNT          ; ELSE, CHANGE THE CONTROLLER DEFINITION
    252 000566  [illegible]06                               BCS     40$             ; IF CS, ERROR
    253 000570  032767  000000G 000000G 20$:                BIT     #OL.UNT,$OPRND ; CHANGE THE UNIT/STATION DEFINITION ?
    254 000576  001402                                      BEQ     40$             ; IF EQ, NO
    255 000600  004767  000000G                             CALL    $SEUNT          ; ELSE, CHANGE THE UNIT DEFINITION
    256 000604  032767  000000G 000000G 40$:                BIT     #OL.X25,$OPRND  ; ANY X.25 LEVEL 2 PARAMETERS ?
    257 000612  001402                                      BEQ     50$             ; IF EQ, NO
    258 000614  004767  000000G                             CALL    $SEX2P          ; ELSE UPDATE X2P$DF
    259 000620  000207                          50$:        RETURN
```

```
        .TITLE  CFEDSP - DIPATCH ROUTINE
        .IDENT  /V05.00/

;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; CFE - DISPATCH ROUTINE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
**FILE**ID**CFEIMP





  CCCCCCCC      FFFFFFFFFF       EEEEEEEEEE         IIIIII        MM        MM    PPPPPPPP
  CCCCCCCC      FFFFFFFFFF       EEEEEEEEEE         IIIIII        MM        MM    PPPPPPPP
CC              FF               EE                   II          MMMM    MMMM    PP        PP
CC              FF               EE                   II          MMMM    MMMM    PP        PP
CC              FF               EE                   II          MM   MM   MM    PP        PP
CC              FF               EE                   II          MM   MM   MM    PP        PP
CC              FFFFFFFF         EEEEEEEE             II          MM        MM    PPPPPPPP
CC              FFFFFFFF         EEEEEEEE             II          MM        MM    PPPPPPPP
CC              FF               EE                   II          MM        MM    PP
CC              FF               EE                   II          MM        MM    PP
CC              FF               EE                   II          MM        MM    PP                    ....
CC              FF               EE                   II          MM        MM    PP                    ....
  CCCCCCCC      FF               EEEEEEEEEE         IIIIII        MM        MM    PP                    ....
  CCCCCCCC      FF               EEEEEEEEEE         IIIIII        MM        MM    PP                    ....




LL                 SSSSSSSS      TTTTTTTTTT
LL                 SSSSSSSS      TTTTTTTTTT
LL               SS                   TT
LL               SS                   TT
LL               SS                   TT
LL               SS                   TT
LL                 SSSSSS             TT
LL                 SSSSSS             TT
LL                         SS         TT
LL                         SS         TT
LL                         SS         TT
LL                         SS         TT
LLLLLLLLLL       SSSSSSSS             TT
LLLLLLLLLL       SSSSSSSS             TT
```

```
   458                             ;
   459 000020                               .ASECT
   460             000000                   .=0
   461 000000                               .BLKW   1               ; C.LNK
   462 000002                               .BLKW   1               ; C.STS
   463 000004                      P$SNAM::.BLKW    2               ; RAD50 NETWORK NAME
   464 000010                      P$SPOR::.BLKW    1               ; NUMBER OF PORTS SUPPORTED (1-256)
   465 000012                      P$SFLG::.BLKW    1               ; FLAGS
   466 000014                      P$STLO::.BLKW    1               ; TRANSPORT SUBADDRESS LOW
   467 000016                      P$STHI::.BLKW    1               ; TRANSPORT SUBADDRESS HIGH
   468             000020          P$SLEN==.                                ; LENGTH OF THE BLOCK
   469 002054                               .PSECT
   470
   471                             ;*****
   472                             ;
   473                             ; X.25 LEVEL 2 DESCRIPTOR BLOCK
   474                             ;
   475 000020                               .ASECT
   476             000000                   .=0
   477 000000                               .BLKW   1               ; C.LNK
   478 000002                               .BLKW   1               ; C.STS
   479 000004                      X$2LIN::.BLKW    1               ; LINE ID: RAD50 PROCESS NAME
   480 000006                               .BLKW   2               ; CONTROLLER, UNIT NUMBERS
   481 000012                      X$2WSZ::.BLKW    1               ; WINDOW SIZE (0-7)
   482 000014                      X$2BSZ::.BLKW    1               ; BLOCK SIZE
   483 000016                      X$2RTC::.BLKW    1               ; RETRANSMIT COUNT
   484 000020                      X$2RTT::.BLKW    1               ; RETRANSMIT TIMER VALUE
   485 000022                      X$2HBT::.BLKW    1               ; HOLDBACK TIMER VALUE
   486             000024          X$2LEN==.
   487 002054                               .PSECT
   488
   489                             ;*****
   490                             ;
   491                             ; DEFINE THE PVC PARAMETER BLOCK
   492                             ;
   493 000024                               .ASECT
   494             000000                   .=0
   495 000000                               .BLKW   1               ; C.LNK
   496 000002                               .BLKW   1               ; C.STS
   497 000004                      P$VNAM::.BLKW    2               ; RAD50 NAME
   498 000010                      P$VLCN::.BLKW    1               ; LOGICAL CHANNEL NUMBER
   499 000012                      P$VCT:: .BLKW    1               ; COUNTER TIMER
   500 000014                      P$VPSZ::.BLKW    1               ; MAXIMUM BLOCK SIZE
   501 000016                      P$VWSZ::.BLKW    1               ; WINDOW SIZE
   502 000020                      P$VOWN::.BLKW    1               ; RAD50 OWNER PROCESS
   503 000022                      P$VFLG::.BLKB    1               ; FLAGS BYTE
   504 000023                      P$VSTA::.BLKB    1               ; =0 --> OFF, >0 --> ON
   505             000024          P$VLEN==.
   506 002054                               .PSECT
   507
   508                             ;*****
   509                             ;
   510                             ; DEFINE THE LOCAL DTE PARAMETER BLOCK
   511                             ;
   512 000024                               .ASECT
   513             000000                   .=0
   514 000000                               .BLKW   1               ; C.LNK
```

```
                                .SBTTL  MACRO DEFINITIONS

                        ;
                        ; LIBRARY MACROS
                        ;
                                .IF NDF I$$AS
                                .MCALL  DIR$,RUN$,EMSG$,EMSG$R,NTLER$,NHWDF$,RETC,SAVMAP,RESMAP
                                .MCALL  ALUN$,VCBDF$,HDRDF$,DCBDF$,UCBDF$,NSFDF$,PDVDF$

 000000                         NHWDF$                          ;HARDWARE REGISTER DEFINITIONS
 000000                         PDVDF$                          ;DEFINE PDV TABLE OFFSETS
 000000                         NSFDF$                          ;DEFINE NETWORK FLAGS
 000000                         VCBDF$                          ;DEFINE VCB OFFSETS
 000000                         DCBDF$                          ;DCB DEFINITIONS
 000000                         UCBDF$                          ;UCB DEFINITIONS
 000000                         HDRDF$                          ;TASK HEADER OFFSETS
```

```
                                  .SBTTL  DATA



                          ;
                          ; AT THE MOMENT, ASSUME THAT IF MEMORY MANAGEMENT IS IN EFFECT, THAT
                          ; THE MACHINE IS AN 11/70 (ONLY AFFECTS UMR LOADING CODE).
                          ;

                          ;
                          ; LIBRARY MACROS
                          ;
                                  .MCALL  DIR$,CLEF$S,WTSE$S,RQST$,SDAT$,EMSG$R,NTLER$
                                  .MCALL  NHWDF$,PDVDF$,DHBDF$,SLTDF$,CCBDF$

                                  .IF DF  R$$MPL                                                  ;[MP01]
                                  .MCALL OLRDF$,QIOW$S,ALUN$S                                     ;[MP01]
                                  OLRDF$                                                          ;[MP01]
                                  .ENDC                                                           ;[MP01]

000000                            CCBDF$                          ;DEFINE CCB OFFSETS
000000                            SLTDF$                          ;DEFINE SLT OFFSETS
000000                            NHWDF$                          ;HARDWARE REGISTER DEFINITIONS
000000                            PDVDF$                          ;DEFINE PDV OFFSETS
000000                            DHBDF$                          ;DEFINE DECNET HOME BLOCK OFFSETS

          000000          M$$EXT == 0                             ; ASSUME UMR SUPPORT

                          ;
                          ; LOCAL (IMPURE) DATA
                          ;
000000                            .PSECT  DATA,D

                          ;
                          ; ERROR MESSAGES
                          ;
                                  .IF     NDF,R$$11S
                                  .ENABL  LC
                          NTLER$  ,10,NONE,$NIERR,.,<Clearing System>
                                  .DSABL  LC
                                  .ENDC

000000    000030          INIFUN: .WORD   FS.INI!FC.PCT           ; PROCESS INIT FUNCTION CODE

                                  .IF DF  R$$MPL                                                  ;[MP01]
                          STATUS: .BLKW   2                       ; I/O STATUS FOR HRC... INIT    ;[MP01]
                          NTLER$  ,11,NONE,$NIERR,,,<HRC... Reconfiguration Failure>              ;[MP01]
                                  .ENDC                                                           ;[MP01]

                          ;
                          ; START OF PURE CODE
                          ;
000000                            .PSECT
```

```
NTICSR  MACRO V05.03b Saturday 29-Jun-85 03:30
Table of contents
```

```
     6-   54     DATA
     7-   96     CSR CHECK
     55                                          .SBTTL  DATA
     56
     57                                   ;
     58                                   ; LIBRARY MACROS
     59                                   ;
     60                                          .MCALL  DHBDF$,NHWDF$
     61 000000                                   DHBDF$                          ;DEFINE DECNET HOME BLOCK OFFSETS
     62 000000                                   NHWDF$                          ;DEFINE THE HARDWARE REGISTERS
     63
     64
     65
     66                                   ;
     67                                   ; LOCAL EQUIV'S
     68                                   ;
     69         176000                    AREA    = 176000                        ;AREA MASK
     70
     71
     72                                   ;
     73                                   ; START OF PURE CODE
     74                                   ;
     75 000000                                   .PSECT
```

```
 000132  032701  000100                    BIT     #NF.LV2,R1              ;* LEVEL 2 ROUTER ?
 000136  001404                            BEQ     40$                     ;* IF NO - BRANCH

 000140  012703  000004'                   MOV     #RCP2NM,R3              ;* POINT TO TASK NAME
 000144                                    CALL    @SRSTD                  ;* IS LEVEL 2 ROUTING CONTROL PR
 000150  006166  000004           40$:     ROL     4(SP)                   ;* SAVE CARRY ...
 000154  006366  000004                    ASL     4(SP)                   ;* IN 2ND BIT
 000160                           50$:     RETURN                          ;* LEAVE SYSTEM STATE

 000162  006001                   60$:     ROR     R1                      ; SET CARRY IF NS IS NOT INSTALLED
 000164  103003                            BCC     70$                     ; IF NOT SET - CONTINUE
 000166                                    EMSG$R  15                      ; NS.... NOT INSTALLED
 000174  006001                   70$:     ROR     R1                      ; SET CARRY IF RCP(1 OR 2) IS NOT INSTALLED
 000176  103003                            BCC     80$                     ; IF NOT SET - CONTINUE
 000200                                    EMSG$R  19                      ; RCP(1 OR 2) NOT INSTALLED
 000206  016702  000000G          80$:     MOV     $DBXPT,R2               ; GET ADDRESS OF XPT'S DDB

 000212                           90$:     .IF NDF R$$11S

                                           TST     N$PLD+2(R2)             ; IS THERE A PLD DATA BASE ?
                                           BNE     150$                    ; IF NE, YES
                                           MOV     @DECPT,R1               ; GET DECNET HOME BLOCK ADDRESS
                                           MOV     D$NLN(R1),R0            ; NUMBER OF NSP CHANNELS
                                           MOV     R0,N$PLD(R2)            ; STORE # OF CHANNELS IN NSP'S DB
                                           MOV     #P$LEN,R1               ; GET SIZE OF EACH PLD
                                           CALL    $MUL                    ; CALCULATE SIZE OF PLD'S
                                           MOV     N$PLD(R2),R0            ; GET NUMBER OF CHANNELS * 2
                                           ASL     R0                      ; ...
                                           ADD     R0,R1                   ; UPDATE NUMBER OF BYTES NEEDED
                                           CLR     R0                      ; ASSUME FAILURE

                                           SWSTK$  120$                    ;* ENTER KERNEL MODE
                                           CALL    @ALOCB                  ;* TRY TO ALLOCATE THE PLD DB
                                           BCS     110$                    ;* IF CS, FAILURE
                                           MOV     R0,2(SP)                ;* ELSE, RETURN ADDRESS IN USER R0
                                           MOV     @DECPT,R1               ;* GET DECNET HOME BLOCK ADDRESS
                                           MOV     D$NLN(R1),R4            ;* GET NUMBER OF CHANNELS
                                           MOV     R4,R1                   ;* GET NUMBER OF CHANNELS * 2
                                           ASL     R1                      ;* ...
                                           MOV     R0,R3                   ;* POINT TO PLD ALLOCATION
                                           ADD     R1,R0                   ;* POINT PAST PLD VECTOR
                                  100$:    MOV     R0,(R3)+                ;* SET UP PLD VECTOR
                                           ADD     #P$LEN,R0               ;* POINT TO NEXT ENTRY IN PLD
                                           DEC     R4                      ;* ALL CHANNELS PROCESSED?
                                           BGT     100$                    ;* BR IF NO

                                  110$:    RETURN                          ;* BACK TO USER MODE

                                  120$:    MOV     R0,N$PLD+2(R2)          ; SAVE PLD ADDRESS
                                           BNE     130$                    ; IF SUCCESFUL ALLOCATION - BRANCH
                                           EMSG$R  18                      ; PLB ALLOCATION FAILURE

                                  130$:    MOV     R0,PLDADR               ; SAVE PLB ADDRESS AGAIN
                                           MOV     R1,PLDLEN               ; AND LENGTH OF THE PLD DATA BASE
                                           ASR     R1                      ; CONVERT BYTES TO WORDS
                                           MOV     N$PLD(R2),-(SP)         ; GET NUMBER OF CHANNELS * 2
                                           SUB     (SP),R1                 ; DON'T ZERO PLD VECTOR
```

Table of contents

```
     6-   54    DATA
     7-   98    MARKED FOR ENABLE SCAN
     8-  255    ENABLE A LINE
     9-  310    DLMSET - SET UP COUNTER BLOCK FIELDS FOR DLM CIRCUITS
```

```
NTIMFE11S  CREATED BY  MACRO  ON 24-JUL-85 AT 15:52      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE             REFERENCES

LN.CLO  = 000000          #6-74
LN.DUM  = 000005          #6-74
LN.LOA  = 000004          #6-74
LN.LOO  = 000003          #6-74
LN.OAU  = 000003          #6-74
LN.OFF  = 000001          #6-74
LN.ON   = 000000          #6-74        7-202      7-244
LN.OOP  = 000004          #6-74
LN.OPE  = 000001          #6-74
LN.REF  = 000002          #6-74
LN.SER  = 000002          #6-74
LN.STA  = 000017          #6-74
LN.SUB  = 000360          #6-74
LN.TRI  = 000006          #6-74
L.COST    000015          #6-74
L.CTL     000012          #6-74        7-166
L.CVA     177776          #6-74
L.DDM     000002          #6-74
L.DDS     000004          #6-74
L.DLC     000003          #6-74        7-129
L.DLM     000006          #6-74
L.DLS     000010          #6-74
L.FLG     000000          #6-74        7-125      7-132      7-135     *7-210      7-214     *7-236     *7-243
L.KRBA    000016          #6-74
L.LEN   = 000022          #6-74
L.MPF     000022          #6-74        7-137      7-168
L.NMST    000020          #6-74       *7-244
L.NSTA    000014          #6-74        7-158
L.OWNR    000021          #6-74
L.UNT     000013          #6-74
M$$EXT  = 000000          #6-60
M$$MGE  = 000000            6-60
NF$MOU  = 000040           7-183       7-227
N$FLG     000005           7-183       7-227
N$$VCT  = ******           8-299
PDVTA   = ******    GX     7-130       7-177      7-221      8-275
R$$MPL  = ******           7-143
R$$11D  = ******            6-77
R$$11M  = 000000            6-77
R$$11S  = 000001          #4-2         6-77       8-279      9-327
SF.ACT  = 000200          #6-74        7-137      7-201
SF.ENA  = 000100          #6-74        7-169      7-192
SF.LPB  = 000004          #6-74
SF.MFL  = 000040          #6-74
SF.PAC  = 000020          #6-74
SF.REA  = 000010          #6-74
SF.SER  = 000001          #6-74
SF.SVC  = 000002          #6-74
SF.UNL  = 000040          #6-74
SLTMA   = ******    GX     7-122
SLTNM   = ******    GX     7-120
STAADD    000000 R        #6-90       *7-124     *7-197
```

```
                                        ; THE ROUTINE NTIRCX AND FROM ABOVE
        MOV     #$NIRTN,R0              ;POINT AT SENDING TASK'S NAME
        MOV     (R0)+,NTISEN+S.DATN     ;SAVE SENDER'S NAME IN
        MOV     (R0)+,NTISEN+S.DATN+2   ;  RETURN DIRECTIVE
        MOV     (R0)+,$NIFNC            ;SAVE FUNCTION CODE
        MOV     (R0)+,$NIFNC+2          ; SAVE TWO WORDS OF SENDER'S DATA
        MOV     (R0),$NIFNC+4           ; (USED FOR POWERFAIL RECOVERY)

        CLR     $NIAST                  ;ASSUME CALLER NOT NOTIFIED VIA AST
        BIT     #LS.AST,$NIFNC          ;NOTIFY CALLER VIA AST?
        BEQ     20$                     ;BR IF NO
        MOV     LR.AST+$NIRDB,$NIAST    ;STORE ADDRESS OF NOTIFICATION AST
        MOV     LR.PRM+$NIRDB,$NIPRM    ;STORE NOTIFICATION AST PARAMETER
20$:
        .ENDC ;DF S$$BAS
                                        ;FALL THROUGH
                                        ;INTO MAIN LOOP .........
```

```
    829                                         .ENDC
```

Table of contents

```
     6-   55     DATA
     7-   80     START CEX TIMERS
     8-  121     T1SIN - INSERT 1 SECOND TIMER FORK BLOCK
     8-  122     T50IN - INSERT 50 MSEC TIMER FORK BLOCK
     9-  156     CLINS - INSERT ENTRY ON CLOCK QUEUE
```

```
                                        .SBTTL  DATA

                                ;
                                ; LIBRARY MACROS
                                ;
                                        .MCALL  SCBDF$

        000000                          SCBDF$                  ; UMR BLOCK OFFSETS


                                ;
                                ; START OF PURE CODE
                                ;
        000000                          .PSECT
```

```
    172                                         .SBTTL  $CLERR - CLEAR ERROR MESSAGE ADDRESS
    173                                 ;+
    174                                 ; INPUTS:       NONE
    175                                 ;
    176                                 ; OUTPUT:       $ERROR = 0
    177                                 ;-
    178 000760                          $CLERR::
    179 000760  005067  177522                  CLR     $ERROR          ; NO MORE ERROR
    180 000764  000207                          RETURN
    181
    182         000001                          .END
```

```
                                                .SBTTL  SETCIR - SET CIRCUIT PARAMETERS
                                                .SBTTL  SETKNC - SET KNOWN CIRCUIT PARAMETERS
                                        ;+
                                        ; *** - SETCIR - SET CIRCUIT PARAMETERS
                                        ; *** - SETKNC - SET KNOWN CIRCUIT PARAMETERS
                                        ;
                                        ; INPUT:
                                        ;       $OPRND = OPERANOS PRESENT
                                        ;
                                        ; OUTPUT:
                                        ;       CONTROL PASSED TO PROPER ROUTINE
                                        ;
                                        ;-
                                                .ENABL  LSB

000622  032767  000000G 000000G SETCIR: BIT     #WC.CIR,$WILD   ; ANY WILDCARDS ?
000630  001416                          BEQ     30$             ; IF EQ, NO - ELSE APPLY "KNOWN" CHECKS
                                        ;
                                        ; WILDCARD CIRCUIT ID: ONLY ALLOW PARAMETERS COMMON TO ALL CIRCUITS
                                        ;
000632                                  SETKNC:
000632  032767  000000G 000000G                 BIT     #OC.NWC,$OPRND  ; CHECK FOR BAD WILDCARD PARAMS
000640  001054                                  BNE     60$             ; IF NE, ERROR
000642  004767  000242                          CALL    SETPVC          ; SET PVC PARAMETERS
000646  004767  000244                          CALL    SETSVC          ; APPLY CHANGES TO SVCS
                                        ;
                                        ; CLEAR OPTION VALUES WHICH ARE SHARED BETWEEN PSI CIRCUIT
                                        ; AND LINES (UNT$DF AND SLT$DF).
                                        ;
000652  042767  000000C 000000G                 BIC     #^C<OC.CST!OC.HTM!OC.STA!OC.APR!OC.MXR!OC.RPR>,$OPRND
000660  004767  000144                          CALL    SETTRN          ; LIKEWISE FOR TRANSPORT CIRCUITS
000664  000460                                  BR      90$             ; EXIT ROUTINE
                                        ;
                                        ; CIRCUIT ID SPECIFIED: DETERMINE CIRCUIT TYPE
                                        ;
000666  012700  000000C                 30$:    MOV     #OC.SVC!OC.PVC!OC.CTM!OC.DTE!OC.MBL!OC.MWI,R0 ; ASSUME TRANSPORT CIRCUIT
000672  012701  001030'                         MOV     #SETTRN,R1      ; POINT TO TRANSPORT ROUTINE
000676  032767  000000G 000000G                 BIT     #FL.X25,$FLAGS  ; PARSED CIRCUIT AN X.25 CIRCUIT ?
000704  001426                                  BEQ     40$             ; IF EQ, OWNER = TRANSPORT
000706  005767  000000G                         TST     $PSIPT          ; PSI IN SYSTEM ?
000712  001441                                  BEQ     80$             ; IF EQ, NOT A PSI SYSTEM
000714  032767  000000G 000000G                 BIT     #FL.CRE,$FLAGS  ; CREATE A VIRTUAL CIRCUIT ?
000722  001403                                  BEQ     35$             ; IF EQ, NO
000724  052767  000000G 000000G                 BIS     #FL.PVC,$FLAGS  ; CREATE A PVC
000732  012700  000000C                 35$:    MOV     #OC.SVC!OC.APR!OC.TRI,R0 ; ASSUME PVC
000736  012701  001110'                         MOV     #SETPVC,R1      ; ...
000742  032767  000000G 000000G                 BIT     #FL.PVC,$FLAGS  ; PARSED CIRCUIT A PVC ?
000750  001004                                  BNE     40$             ; IF NE, YES
000752  012700  000000C                         MOV     #OC.PVC!OC.APR!OC.TRI,R0 ; ASSUME SVC
000756  012701  001116'                         MOV     #SETSVC,R1      ; ...

                                        ;
                                        ; CHECK FOR ILLEGAL PARAMETER GROUPING AND DISPATCH TO OPERATE ROUTINE
                                        ;
000762  030067  000000G                 40$:    BIT     R0,$OPRND       ; CHECK FOR ILLEGAL GROUPING
000766  001006                                  BNE     70$             ; IF NE, ILLEGAL PARAMETER GROUPING
000770                                          CALLR   (R1)            ; DISPATCH TO ROUTINE
```

```
                                   ;****
                                   ; LOCAL DATA
                                   ;****

                                   ;
                                   ; DEFINE DISPATCH TABLE
                                   ;
 000000   000000G                  DSPTBL: .WORD   $CFDEF          ; ''DEFINE'' COMMAND
 000002   000000G                          .WORD   $CFLIS          ; ''LIST'' COMMAND
 000004   000000G                          .WORD   $CFHLP          ; ''HELP'' COMMAND
 000006   000000G                          .WORD   $FIN            ; ''EXIT'' COMMAND
 000010   000000G                          .WORD   .EXIT           ; ''KILL'' COMMAND
 000012   000000G                          .WORD   $CFPRG          ; ''PURGE'' COMMAND
```

Table of contents

```
     5-   64     SYMBOL DEFINITIONS
     6-   89     DATA DEFINITIONS FOR FILE I/O
     7-  123     GLOBAL DATA
     8-  327     BIT DEFINITIONS AND FUNCTION CODES
     9-  344     DEFINE TEMPLATES GLOBALLY
    13- 1204     ERROR MESSAGES
```

B	1	NTIUMR
C	1	
D	1	
E	1	
F	1	
G	1	
H	1	
I	1	
J	1	
K	1	NTIX25
L	1	
M	1	
N	1	
B	2	
C	2	
D	2	
E	2	
F	2	
G	2	
H	2	
I	2	
J	2	
K	2	NTICEX
L	2	
M	2	
N	2	
B	3	
C	3	
D	3	
E	3	
F	3	
G	3	
H	3	
I	3	
J	3	
K	3	
L	3	
M	3	NTICSR
N	3	
B	4	
C	4	
D	4	
E	4	
F	4	
G	4	
H	4	
I	4	
J	4	
K	4	NTIDLI
L	4	
M	4	
N	4	
B	5	
C	5	
D	5	
E	5	
F	5	
G	5	NTILON
H	5	
I	5	
J	5	
K	5	
L	5	
M	5	
N	5	
B	6	
C	6	
D	6	
E	6	
F	6	
G	6	
H	6	
I	6	
J	6	
K	6	
L	6	
M	6	NTIMFE
N	6	
B	7	
C	7	
D	7	
E	7	
F	7	
G	7	
H	7	
I	7	
J	7	
K	7	
L	7	
M	7	
N	7	
B	8	
C	8	
D	8	
E	8	NTINIT
F	8	
G	8	
H	8	
I	8	
J	8	
K	8	
L	8	
M	8	
N	8	
B	9	
C	9	
D	9	
E	9	
F	9	
G	9	
H	9	
I	9	
J	9	
K	9	
L	9	
M	9	
N	9	
B	10	
C	10	
D	10	
E	10	
F	10	
G	10	
H	10	
I	10	
J	10	
K	10	
L	10	
M	10	NTITIM
N	10	
B	11	
C	11	
D	11	
E	11	
F	11	
G	11	
H	11	
I	11	
J	11	
K	11	NTIUMR
L	11	
M	11	
N	11	
B	12	
C	12	
D	12	
E	12	
F	12	
G	12	
H	12	CFECOM
I	12	
J	12	
K	12	
L	12	
M	12	
N	12	
B	13	
C	13	
D	13	
E	13	CFEDEF
F	13	
G	13	
H	13	
I	13	
J	13	
K	13	
L	13	
M	13	
N	13	
B	14	
C	14	
D	14	
E	14	
F	14	
G	14	
H	14	
I	14	
J	14	
K	14	CFEDSP
L	14	
M	14	
N	14	
B	15	
C	15	
D	15	
E	15	CFEHLP
F	15	
G	15	
H	15	
I	15	
J	15	
K	15	
L	15	
M	15	CFEIMP
N	15	
B	16	
C	16	
D	16	
E	16	
F	16	
G	16	
H	16	
I	16	
J	16	
K	16	
L	16	
M	16	